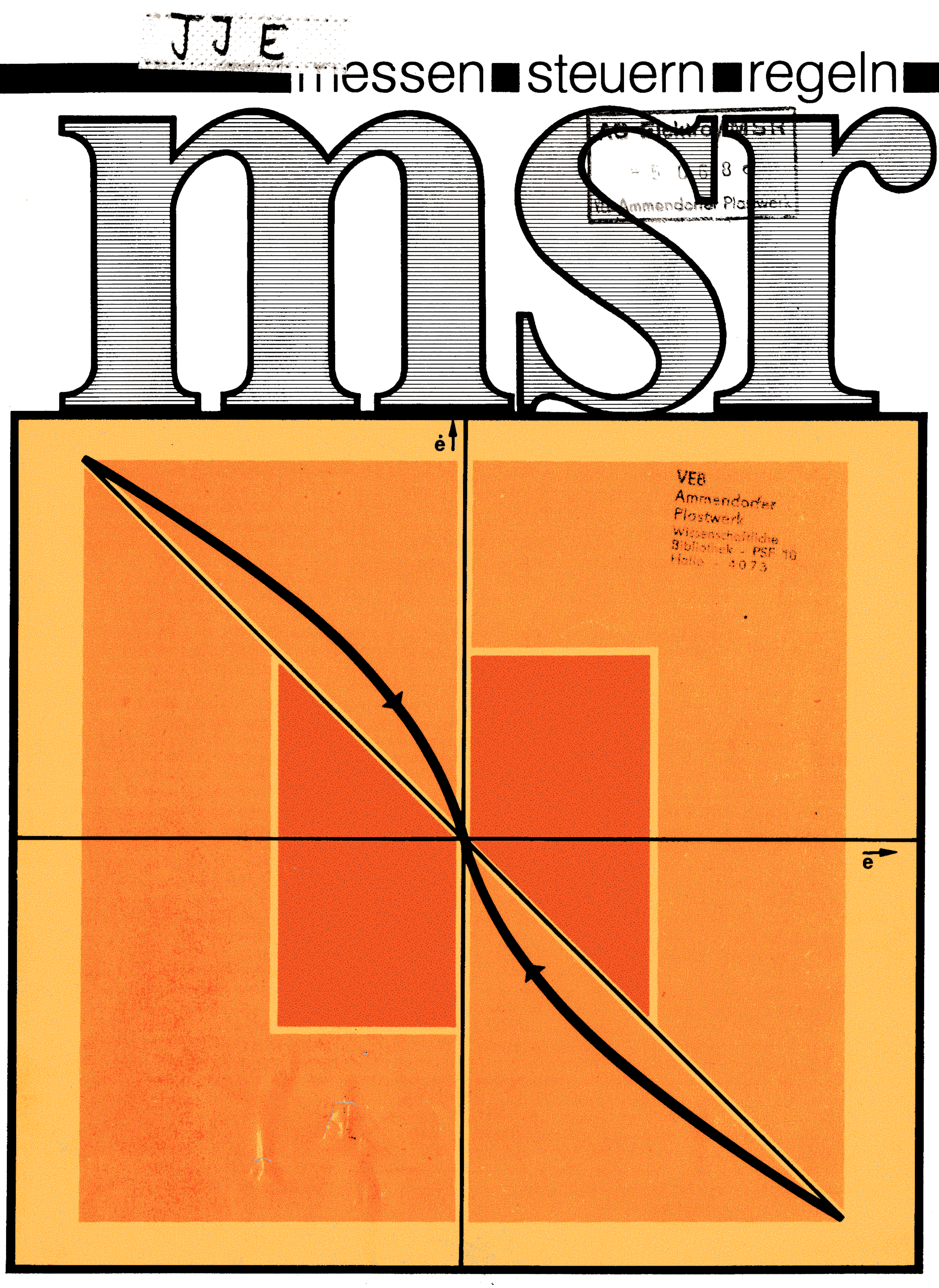
messen steuern regeln



VT
VEB Verlag Technik Berlin
EVP: 4,– M
ISSN 0026-0347

5/89



20 Jahre ZKI der AdW

Die wissenschaftlich-technische Revolution wird heute im wesentlichen von der Entwicklung und umfassenden Anwendung moderner Informationstechnologien geprägt. Weltweit werden große Anstrengungen unternommen, um neue Technologien der Informationsgewinnung, -speicherung, -verarbeitung und -verwertung zu schaffen, mit denen ein weiterer Produktivitätsfortschritt erreicht werden soll. Neue Wirkprinzipien von Informationstechnologien sind immer mehr aus der Sicht des menschlichen Denkens gefragt, so daß diese Technologien zunehmend auch schöpferische („intellektuelle") Prozesse des Menschen unterstützen oder auch ersetzen können.
Als vor zwanzig Jahren — mit Wirkung vom 1. Mai 1969 — das Zentralinstitut für Kybernetik und Informationsprozesse (ZKI) als Forschungseinrichtung der damaligen Deutschen Akademie der Wissenschaften zu Berlin gegründet wurde, stand vor dem Kollektiv von rund 200 Mitarbeitern die Aufgabe, „wissenschaftliche Grundlagen für die Automatisierung der technischen Produktionsvorbereitung" zu erarbeiten. Unter der Leitung seines ersten Direktors (von 1969 bis 1977), Prof. Dr. rer. nat. habil. HORST VÖLZ, wurden im Rahmen der Forschungskomplexe Grundlagen der Kybernetik, Technische Kybernetik, Magnetomotorische Speicher und Kognitive Psychologie volkswirtschaftlich wichtige Aufgabenstellungen bearbeitet, die dann gegen Ende der 70er Jahre durch weitere Aufgaben zur Bildverarbeitung, zur Systemanalyse und Simulation, zum VLSI-Schaltkreisentwurf sowie zur Anwendung von Verfahren der Rechentechnik in der nichtkonventionellen Automatisierungstechnik, insbesondere zur Schaffung interaktiver CAD-Arbeitsplätze, ergänzt wurden.
Heute gehört das ZKI, das seit 1977 von Prof. Dr. sc. nat. VOLKER KEMPE, Ordentliches Mitglied der AdW, geleitet wird, zu den leistungsfähigsten Forschungseinrichtungen der Akademie der Wissenschaften der DDR und ist ein anerkannter Kooperationspartner solcher wichtigen Industriekombinate, wie der VEB Carl Zeiss JENA, das Kombinat Robotron, das Kombinat „7. Oktober" Berlin und das Werkzeugmaschinenbaukombinat „Fritz Heckert" Karl-Marx-Stadt. Im Rahmen der Forschungen zur Informatik und Kybernetik werden Beiträge zur Entwicklung der fachspezifischen sowie geräte- und programmtechnischen Grundlagen der Rechentechnik und ihrer Anwendung geleistet. Forschungen zur Beschreibung und zum Entwurf diskreter Systeme beinhalten die Entwicklung von Methoden und Programmen für die Projektierung von Rechnern und hoch- bzw. höchstintegrierten Schaltkreisen. Die Forschungsarbeiten zur künstlichen Intelligenz umfassen Algorithmen, Programme und gerätetechnische Realisierungen für die zwei- und dreidimensionale visuelle Erkennung sowie Beiträge zur automatischen Handlungsplanung und zu natürlichsprachigen Abfragesystemen. Für den Aufbau von Expertensystemen erfolgt eine zunehmende Konzentration auf Inferenztechniken, Wissensrepräsentationsmethoden und KI-Sprachen, die auch für künftige Rechnergenerationen unentbehrlichen Vorlauf darstellen.
Das Forschungsprofil des ZKI hat sich zunehmend von einer Orientierung auf Anwendungen der Rechentechnik zur Entwicklung der Kernkomponenten der Rechentechnik selbst verlagert. Dementsprechend bilden die Forschungsrichtungen zur Untersuchung von Rechnerarchitekturen, zur Computergrafik und Bildverarbeitung sowie zur künstlichen Intelligenz zusammen mit der Entwicklung neuer Entwurfssysteme für höchstintegrierte Schaltkreise und Rechner die Hauptlinie der weiteren Entwicklung des Instituts, um die herum Anwendungslinien der Rechentechnik, dem Schwerpunkt der Produktionsautomatisierung, verfolgt werden.
Das Hauptanliegen der gegenwärtig eingeleiteten wissenschaftlich-technischen Revolution ist die Steigerung der Leistungskraft der Volkswirtschaft; nur dadurch kann die weitere kontinuierliche Verbesserung des gesamten Lebensniveaus der Menschen erreicht werden. Solide Basis dieser Entwicklung ist u. a. die industrielle Automatisierung, gekennzeichnet durch eine durchgängige rechnergestützte Arbeitsweise — vom Entwurf über die Konstruktion bis hin zur Planung der Maschinenbelegung und der aktuellen Steuerung einzelner Maschinen bzw. des gesamten Fertigungssystems einschließlich der notwendigen TUL-Prozesse. Die Einführung der rechnergestützten Konstruktion ist am effektivsten, wenn sie papierlos in die Produktion mündet. CAM ist das Kennzeichen für die rechnergestützte Produktion, die nicht nur die numerische Steuerung einzelner Maschinen oder Maschinengruppen, sondern vor allem komplexe Probleme der technologischen Arbeitsorganisation umfaßt, durch deren Lösung eine wesentliche Effektivitätssteigerung erreicht werden soll.
Die Verknüpfung und durchgängige Anwendung von CAD und CAM ist jedoch — auch international — noch nicht alltäglich. Im Mittelpunkt der Forschungen des ZKI zur Fertigungs- und Prozeßautomatisierung stehen deshalb Arbeiten für CAD/CAM zur flexiblen Automatisierung, die sich auf die Schaffung von Hilfsmitteln (insbesondere Simulationssystemen) für den Entwurf flexibler Fertigungssysteme sowie auf die Entwicklung von Algorithmen und Programmen zu ihrer organisatorischen Steuerung konzentrieren. Darüber hinaus werden Beiträge zur dezentralen Regelung großer technischer Systeme (z. B. das Elektroenergie-Verbundsystem der DDR), zum rechnergestützten Entwurf von Reglern sowie zur Entwicklung von Mehrmikrorechnersystemen geleistet. Im Rahmen der Arbeiten zur Schaffung interaktiver CAD-Arbeitsplätze werden neben Systemen für den rechnergestützten Entwurf von mikroelektronischen Schaltkreisen und Automatisierungsanlagen auch Systeme und Anwendungslösungen der automatisierten digitalen Bildverarbeitung sowie Computergrafikpakete entwickelt. Forschungsarbeiten zur magnetomotorischen Speicherung beinhalten Grundlagenuntersuchungen zur digitalen Dichtspeichertechnik im Hinblick auf die Entwicklung neuer Speichersysteme, wie sie vornehmlich auch bei Weltraummissionen zum Einsatz kommen.
Die Einführung moderner Informationstechnologien wird nicht auf die Produktionssphäre beschränkt bleiben, sondern alle Bereiche unseres gesellschaftlichen Lebens durchdringen. Schwerpunkt der Forschungsarbeiten des ZKI zur angewandten Systemanalyse ist die Entwicklung von Verfahren der rechnergestützten Modellierung und Vorbereitung von Entscheidungsprozessen für komplexe und unvollständig beschriebene Systeme. Die Softwareentwicklungen sind auf Anwendungen in makroökonomischen Planungs- und Leitungsprozessen sowie auf Entscheidungsvorbereitungen in der Land- und Forstwirtschaft sowie im Umweltschutz orientiert. Gegenstand der Forschungen zur kognitiven Psychologie ist die Analyse, Modellierung und Simulation von Prinzipien der Speicherung und Verarbeitung von Information im menschlichen Gedächtnis.
Das Zentralinstitut für Kybernetik und Informationsprozesse beschäftigt heute — in seinem Jubiläumsjahr — rund 600 Mitarbeiter, von denen etwa ein Drittel in einem Institutsteil in Dresden ansässig ist. Die Beiträge in diesem Heft sollen den interessierten Lesern einen kleinen Einblick in das Aufgabenprofil dieser Forschungseinrichtung der Akademie der Wissenschaften der DDR ermöglichen.

Prof. Dr. sc. nat. V. KEMPE,
Direktor des ZKI



messen ■ steuern ■ regeln

Wissenschaftlich-technische Zeitschrift für die Automatisierungstechnik

Herausgeber:
Wissenschaftlich-Technische Gesellschaft für Meß- und Automatisierungstechnik (WGMA) in der KDT

Zum Titelbild:

Es gibt gute Gründe, unscharfe Regler-Konzepte (Fuzzy-Regler) auch für sensorgeführte Robotersteuerungen zu verwenden (Nichtlinearitäten, Schwierigkeiten bei der Einstellung der Reglerparameter konventioneller Regler). Unser Beitrag auf S. 210 behandelt diese Thematik. Dabei wird ein Regler auf Basis der Phasenebene eines Systems 2. Ordnung entworfen.

Titelgrafik: G. SCHWESINGER

5

1989

32. Jahrgang
Mai 1989

ISSN 0026-0347

VEB
VERLAG TECHNIK
BERLIN

Aufsätze

D. Pötschke

Künstliche Intelligenz und 5. Rechnergeneration

0. Einleitung

Als der Jenenser Mathematiker G. Frege (1846 bis 1925) im Jahre 1879 seine „Begriffsschrift" veröffentlichte [1] und darin eine einheitliche Sprache für die exakte Formulierung mathematischer Beweise vorschlug, wird er kaum daran gedacht haben, daß diese sogenannte Prädikatenlogik 1. Stufe nach über 100 Jahren für viele andere Wissensgebiete, wie Geologie, Medizin, Technikwissenschaften usw., eine enorme Bedeutung erlangen sollte.
Im Jahre 1982 wurde in Japan mit der Arbeit an einem Großprojekt begonnen [2], das die Entwicklung von Rechnern der 5. Computergeneration innerhalb von 10 Jahren zum Ziel hat. Für diese „intelligenten" Computer soll die Programmiersprache PROLOG (Programmieren in Logik) als Kernsprache auf Maschinenniveau (!) dienen. PROLOG nutzt ganz entscheidend die Ideen von Freges Beschriffsschrift [3]. Bis dahin war aber noch ein weiter Weg zurückzulegen, auf dem einige Meilensteine mit den Leistungen der Mathematiker A. Church, A. Turing, J. v. Neumann und J. A. Robinson gesetzt wurden.
„Das japanische Programm vollzieht einen radikalen Bruch mit dem Entwicklungskonzept der vier vorhergehenden Rechnergenerationen, der noch nicht abgeschlossenen Von-Neumann-Ära, bei der das Ziel darin bestand, mit einem Minimum an Hardware Systeme maximaler Einfachheit zur effektiven Verarbeitung von Software zu schaffen. Mit der 5. Rechnergeneration treten die funktionellen Eigenschaften der Rechner in den Vordergrund, die Geräte- und Programmtechnik leiten sich daraus ab" [4].
Seit der Entwicklung der ersten Rechner in den 40er Jahren, an der J. von Neumann mit seinem Konzept des sequentiell erarbeitenden Computers entscheidenden Anteil hatte [5], stand die Frage, welche der geistigen Tätigkeiten der Mensch dem Rechner übertragen kann. Zunächst war er nur ein Hilfsmittel zur schnellen Lösung spezieller numerischer Probleme, von dem von Neumann noch glaubte, daß einige wenige Anlagen die Bedürfnisse der USA befriedigen würden. Durch seine Weiterentwicklung qualifizierte sich der Rechner aber in den letzten 40 Jahren von der reinen Datenverarbeitungs- zur potentiellen Wissensverarbeitungsanlage. Daran haben die Forschungen zur künstlichen Intelligenz (KI) einen wachsenden Anteil.
Als Geburtsstunde der KI gilt die „Dartmouth Conference" (1956), wenngleich die einheitlichen Anfänge mit den oben genannten Namen eng verbunden sind. Heute kann man folgende Teilgebiete der KI unterscheiden:

1. Wissensrepräsentation und Inferenzprozesse
2. Lernen und Wissensverarbeitung
3. Expertensysteme
4. Verarbeitung natürlicher Sprache
5. Bildverarbeitung/Bildverstehen
6. KI-Sprachen (vor allem LISP, PROLOG)
7. Intelligente Roboter
8. KI-Architektur
9. Cognitive Science.

Ohne auf den bisher erreichten Stand in der KI-Forschung hier näher einzugehen, muß festgestellt werden, daß die KI vor eine Reihe neuer Aufgaben durch die engere Verflechtung mit der Computerentwicklung gestellt wird. Zugespitzt kann dies durch die Gleichung

Mikroelektronik + KI = FGCS
FGCS Fifth Generation Computer Systems, Rechner der 5. Generation

ausgedrückt werden [2].
Daraus ergibt sich eine ganze Reihe von Fragen:
Welche Methoden stellt die KI zur Verfügung, um den Rechner zu „höheren" Leistungen zu befähigen (Sprache, Wissensrepräsentation, Inferenz usw.)?
Welchen Beitrag vermag sie zur Entwicklung der 5. Rechnergeneration zu leisten?
Ist der jetzige Entwicklungsstand von PROLOG und LISP, zwei typischen KI-Sprachen, befriedigend?
Sind Expertensysteme heute schon intelligent?
Wie können die neuen Möglichkeiten der Parallelisierung der Hardware, der enormen Vergrößerung der internen und externen Speicherkapazität, der Spezialprozessoren (Inferenzmaschinen, Datenflußmaschinen usw.), der Vernetzung von Rechnern und nicht zuletzt die wachsende Zuverlässigkeit für die Aufgaben der KI genutzt werden?
Welche neuen Aufgaben ergeben sich daraus?
Auf einige Fragen soll hier näher eingegangen werden, ohne endgültige Antworten anzustreben. Vielmehr sollen neue Probleme aufgezeigt werden.

1. Drei Triebkräfte der KI

Man kann heute im wesentlichen drei Triebkräfte unterscheiden, die das Entwicklungstempo, die Entwicklungsrichtung und das Investitionsvolumen der KI-Forschung in den einzelnen Ländern bestimmen.

1. Triebkraft: Homo ludens und Cognitive Science
Es besteht ein bemerkenswertes Interesse vieler Wissenschaftler und Programmierer (homo ludens — spielender Mensch) daran, den Computer zu Leistungen zu befähigen, die sonst nur der Mensch erbringen kann.
Die wissenschaftliche Beschäftigung mit diesem Problem hat vor allem KI-Forscher und Psychologen zusammengeführt und trug zur Herausbildung einer neuen Disziplin, der „Cognitive Science", bei. Das Interesse, den Rechner systematisch zu intelligenten Leistungen auf verschiedenen Gebieten zu bringen, führte auch zur Herausbildung der oben aufgeführten Teilgebiete der KI.

2. Triebkraft: Breite Praxiswirksamkeit der KI-Produkte
Eine zweite, wesentliche Triebkraft besteht in der wachsenden, breiten Praxiswirksamkeit bereits entwickelter KI-Produkte, sowohl der KI-Hardware als auch der KI-Software. Diese Techniken werden in wachsendem Maße vor allem in folgenden Gebieten eingesetzt:

— Industrielle Automatisierung
— Büroautomatisierung
— Gesundheitswesen
— Wissenschaft und Technik
— Computerhardware- und -software
— Luftfahrt

Dr. rer. nat. Dieter Pötschke (42) studierte von 1965 bis 1972 Mathematik und Psychologie an der Humboldt-Universität zu Berlin. 1972 Promotion A (Erweiterung der Shannonschen Informationstheorie). Seit 1972 Mitarbeiter am Institut für Kybernetik und Informationsprozesse (ZKI) der AdW der DDR, Bereich „Künstliche Intelligenz".

— Ausbildungswesen
— Militärwesen
— Einzelhandel und Dienstleistungswesen
— Kunst, Kultur und Freizeit.

Dieses Eindringen der KI in die verschiedensten Bereiche von Volkswirtschaft und Wissenschaft hat — auf den internationalen Markt bezogen — eine wachsende Vermarktungsfähigkeit der KI-Produkte zur Folge. Beinahe zu jedem KI-Produkt gibt es heute schon eine oder mehrere nicht unbedeutende Firmen, die dieses Produkt kommerziell vertreiben.
Die britische Firma *SPERRY* schätzt ein, daß der KI-Markt weltweit 1990 ein Volumen von 4 Mrd. Dollar umfassen wird (1984 85 Mill. Dollar) [6]. Hier ist aber das sich rasch verändernde Verhältnis der Ausgaben für Software (S) und Hardware (H) zu berücksichtigen. Nach statistisch signifikanten Untersuchungen in den USA entwickelt sich dort das Verhältnis S/H entsprechend der Boehmschen Kurve [7]. Damit betragen gegenwärtig die Ausgaben für die Software (Entwicklung, Pflege, Wartung) etwa 90% der Gesamtausgaben. Andererseits ermöglichen Spezialprozessoren überhaupt erst die kommerzielle Nutzung von Sprachen wie LISP und Systemen, die in diesen Sprachen implementiert sind. Dies zeigte sich an der Einführung der LISP-Maschinen SYMBOLICS 3600, EXPLORER und CLM von *Texas Instruments*.

3. Triebkraft: Projekte zur Entwicklung der 5. Rechnergeneration

Als wesentliche Triebkraft, die vor allem die Entwicklungsziele und den finanziellen Aufwand für die KI-Forschung bestimmt, haben sich die verschiedenen nationalen und internationalen Projekte zur Entwicklung der 5. Rechnergeneration erwiesen. Diese Projekte beinhalten Teilprojekte, die speziell der Lösung von KI-Problemen gewidmet sind, in sehr unterschiedlichem Maße.
„Das japanische Projekt ist zweifellos das homogenste und ausgewogenste, das mit hohem Risiko versucht, die Basiskomponenten einer neuen Rechnergeneration zur allumfassenden, breitesten Anwendung zu schaffen. Bei weitem weniger homogen und z. T. nur als Antwort auf die sog. japanische Herausforderung entstand eine Reihe anderer Projekte, die deshalb aber nicht unbedingt weniger wirksam sein müssen" [4].
Rechner der 5. Generation sind im wesentlichen durch drei Merkmale gekennzeichnet:

1. Interne Intelligenz
   System zur deduktiven Inferenz; adaptiver Problemlöser; umfangreiche Wissensbasen; höhere Inferenzarten in der Software realisiert (ohne spezielle Hardwarestützung)
2. Intelligentes Mensch-Maschine-Interface
   Eingabe von Lauten und fließender Sprache; Bildeingabe, schnelle Bildvorverarbeitung und leistungsfähige Computergrafik; automatisches Einlesen von Dokumenten (mit sehr hoher Lesegeschwindigkeit); Kommunikation in einer Fachsprache und in eingeschränkter natürlicher Sprache (gesprochen, geschrieben).
3. Hardwareunterstützung der Software durch parallele bzw. konkurrente Abarbeitung
   Inferenzmaschine für die schnelle Abarbeitung von PROLOG; LISP-Prozessoren; Datenflußmaschine; hoher Integrationsgrad (10^7 Transistorfunktionen je Chip); externe Massenspeicher (bis zu 600 Mbyte); hochzuverlässige Architektur; Vernetzung der Rechner auf verschiedenen Niveaus.

Mit diesen Merkmalen soll der Übergang von der Datenverarbeitung zur Wissensverarbeitung, zu den sog. KIPS (Knowledge Information Processing Systems) vollzogen werden.
Damit unterscheidet sich die Grundstruktur eines KIPS grundlegend von Rechnern vorheriger Generationen. Die zentrale Verarbeitungseinheit wird durch die Problemlösungs- und Inferenzmaschine abgelöst, der Hauptspeicher durch die Wissensbasismaschine. An die Stelle der Maschinensprache tritt im japanischen Projekt die logische Programmier- und Wissensrepräsentationssprache PROLOG (im wesentlichen Standard-PROLOG, genauer DEC-10-PROLOG) [8].
Außerdem kann jedes KIPS mit Spezialprozessoren für numerische Rechnungen, Symbolmanipulation sowie Hardware für Bild-, Grafik- und Sprachverarbeitung ausgerüstet werden.
Ein konkretes Ergebnis der 1. Phase (1982 bis 85) des japanischen Projektes FGCS war die Entwicklung eines Prototyps einer PROLOG-Maschine, der sog. PSI-1-Maschine (Personal Sequential Inference), die 1985 in Tokyo und Rotterdam vorgestellt wurde. Die PSI-1-Maschine wurde von der Firma *Mitsubishi* in einer kleinen Serie gebaut und dient ausschließlich Forschungszwecken und der Entwicklung von Software in PROLOG. Zu weiteren Einzelheiten des japanischen Projekts liegen Veröffentlichungen vor [9]. Die geplanten Finanzmittel belaufen sich auf 850 Mill. Dollar, von denen der überwiegende Teil in der dritten Projektphase (1987 bis 1992) von der Industrie aufgewendet wird. Das japanische Projekt setzt ganz bewußt Schwerpunkte in der Forschung bei neuartigen Rechnerarchitekturen, beim Schaltkreisentwurf und bei KI-Problemen, von denen vor allem die logische Programmierung, Wissensrepräsentation, Inferenzprozesse und Expertensysteme betroffen sind.
Antworten auf die japanische Herausforderung stellen in den USA das Projekt SCI (Strategic Computing Initiative) des Verteidigungsministeriums und verschiedene Programme zur Halbleiterforschung, zur Mikroelektronik und Rechentechnik unter starker Betonung der künstlichen Intelligenz sowie der mikroelektronischen Grundlagen dar. Das Projekt SCI wird von der DARPA (Defense Advanced Research Projects Agency), einer Behörde des Department of Defense mit über 600 Mill. Dollar finanziert. Damit soll die militärische Nutzung von Rechnern der 4. und 5. Generation abgesichert werden [10].
Heterogener — den Interessen der beteiligten Länder entsprechend — stellt sich das westeuropäische Projekt ESPRIT (1984 bis 1994) dar, das langfristige Forschungsvorhaben in der fortgeschrittenen Mikroelektronik, der Softwaretechnologie, der fortgeschrittenen Informationsverarbeitung, bei Bürosystemen und in der computerintegrierten Fertigung enthält. Dieses Programm soll die Wettbewerbsfähigkeit Westeuropas in den 90er Jahren auf diesem Sektor sichern.
Auf Initiative der Akademie der Wissenschaften der UdSSR entstand vor vier Jahren ein Konzept zur Schaffung neuer Rechnergenerationen. Es ist das Ergebnis kollektiver Arbeit von Vertretern aller Akademien der sozialistischen Länder, „die sehr wohl die für die Entwicklung der sozialistischen Gesellschaft fruchtbaren Potenzen dieser Technik erkannt haben, aber auch, daß die von den Japanern ausgehende und von den führenden kapitalistischen Industriestaaten angenommene Herausforderung neue Maßstäbe für die Bereitstellung des erforderlichen wissenschaftlichen Vorlaufs in den sozialistischen Ländern setzt" [4]. In der Tafel sind die zehn wissenschaftlichen Komplexprojekte (WKP) dieses Programms aufgeführt. Die Schwerpunkte werden durch die Grundstruktur des erarbeiteten Systemkonzeptes deutlich, dessen Komponenten in zeitlich abgestuften Zielstellungen entstehen. Auch bei diesem Programm stellen vor allem in den wissenschaftlichen Komplexprojekten 1, 3, 8 und 9 Probleme der KI Schwerpunkte der Forschung dar. Es kann hier nicht die Aufgabe sein, näher auf die wichtigen Themen wie Computernetze (WKP4), fehlertolerante Computersysteme (WKP6), Massenspeicher nach neuen Prinzipien (WKP7) und neuartige Architekturen einzugehen. Vielmehr sollen noch auf einige Fragen, die eingangs genannt wurden, eingegangen und die Schwierigkeit und Komplexität der Aufgaben verdeutlicht werden, die im Rahmen der künstlichen Intelligenz als Fachgebiet für die Entwicklung der 5. Rechnergeneration bearbeitet werden müssen.

2. KI-Sprachen und Spezialprozessoren

Bei der Entwicklung eines KI-Systems besteht zunächst das Problem der Wahl einer adäquaten Programmiersprache. Die Entscheidung beim japanischen FGCS-Projekt für PROLOG führte zu einem enormen weltweiten Aufschwung der Forschungen zur logischen Programmierung. Jährlich werden in Japan und in Westeuropa bedeutende internationale Tagungen dazu durchgeführt

[11]. Die PSI-Maschine kann mit etwa 150 KLIPS (LIPS logische Inferenzen je Sekunde; eine logische Inferenz entspricht heute etwa 100 ... 1000 Maschineninstruktionen) PROLOG-Programme sequentiell abarbeiten. Aber gerade die sequentielle Abarbeitung stellt eine Barriere für eine wesentliche Steigerung der Abarbeitungsgeschwindigkeit dar. Als E. SHAPIRO die Konzeption für die Sprache Concurrent PROLOG vorlegte, das auf konkurrierenden Prozessen beruht, entwickelte das für das FGCS-Projekt verantwortliche Institut ICOT einen Compiler für diese Sprache und erarbeitete bald darauf ein entsprechendes Konzept für eine „Parallele Inferenzmaschine" (PIM). Derzeit setzt sich aber für die PIM das Konzept der „guarded Horn clauses" durch [12] bis [14]. Ein Prototyp einer ersten einfachen PIM zur Realisierung der OR-Parallelität, hergestellt von *FUJITSU*, stand dem ICOT seit Ende 1986 zur Verfügung [15]. Erst verschiedene japanische Firmen, die traditionell seit Jahren LISP-Maschinen auf dem Weltmarkt anbieten (wie *FUJITSU*), haben darauf gedrängt, daß in der 2. Phase des FGCS-Projektes auch Forschungen zur schnellen LISP-Abarbeitung vom ICOT aufgenommen oder finanziert wurden.

In den USA hatte dagegen die DARPA für militärische Anwendungen zunächst auf ADA (eine Weiterentwicklung von PASCAL) orientiert. Derzeit ist aber eine Neuorientierung auf COMMON LISP zu beobachten. So finanziert die DARPA ein Programm, in dessen Rahmen die Firma *Texas Instruments* die Entwicklung der CLM (Compact LISP Maschine) als Nachfolger ihrer LISP-Maschine EXPLORER betreibt [16]. Die CLM hat einen 32-bit-Prozessor und 2 Mbyte Hauptspeicher. Im September 1986 wurde ein Prototyp der CLM an die DARPA ausgeliefert. Die Orientierung auf COMMON LISP hat ohne Zweifel eine Reihe von Vorteilen. Dieser Dialekt scheint sich derzeit weltweit als Standard von LISP durchzusetzen.

Aus diesen und anderen Gründen ist es naheliegend, die Vorzüge des funktionalen (LISP) und des logischen (z. B. PROLOG) Programmierstiles zu vereinigen. Einer der ersten Versuche war die Sprache LOGLISP (jetzt SUPERLOGLISP) von J. A. ROBINSON. Eine andere Entwicklungslinie besteht darin, den objektorientierten Programmierstil einzuführen, wie er von den Sprachen SMALLTALK und FLAVOR bekannt ist. Hier ist eine Reihe von objektorientierten Sprachen im Entstehen wie ESP, TS- und TC-PROLOG [17] und [18], COMMON TALK [19] und [20]. Häufig werden logische Programmierung und PROLOG gleichgesetzt. Dies ist aber unzutreffend. Zum einen entspricht PROLOG von der Syntax her nur einem Teil der Prädikatenlogik 1. Stufe, der sog. Horn-Logik. Andererseits besteht für die logische Programmierung nach R. KOWALSKI die Gleichung

$$ALGORITHMUS = LOGIK + KONTROLLSTRATEGIE\,.$$

Danach kann ein Algorithmus in zwei Bestandteile zerlegt werden:

1. einen logischen Teil, der besagt, was der Algorithmus berechnen soll
2. einen Kontrollteil, der besagt, wie der Berechnungsprozeß ablaufen soll.

Es gibt nun eine Reihe von Bemühungen, die Kontrollstrategie bei PROLOG flexibler zu gestalten. Hier sei nur auf METALOG von F. DINCBAS, das Projekt FLEXLOG [21] und verschiedene Versuche verwiesen, durch die Anwendung von Metawissen die Abarbeitung von PROLOG-Programmen effektiver zu gestalten [22], denn derzeit weist PROLOG [8] einige Schwachstellen auf. Man kann die Abarbeitungsstrategie eines derartigen Programms mit einem Schachspieler vergleichen, der die erste Figur, die er noch nicht berührte, auf das nächste freie Feld zieht. Die Aufgabe des Programmierers besteht nun darin, ihm die Figuren so zu präsentieren, daß der Spieler gewinnt.

Das folgende einfache Programm verdeutlicht diese Abarbeitungsstrategie [8]. R sei eine symmetrische, transitive Relation und es gelten $R(a, b)$ und $R(b, c)$, gilt dann $R(c, a)$?

$R(b, a)\,.$
$R(b, c)\,.$
$R(x, y) \colon - R(y, x)\,.$ (Symmetrie)
$R(x, z) \colon - R(x, y), R(y, z)\,.$ (Transitivität)
$?R(c, a)\,.$

Tafel: Programm der Akademien der Wissenschaften der sozialistischen Länder zur Schaffung neuer Generationen von Rechnern [30]

1.	*Wissensverarbeitungssysteme*
1.1	Expertensysteme
1.2.	Grundlagen der Wissensverarbeitung
1.3.	Wissensverarbeitungsmaschinen
1.4.	Verteilte Wissensbasissysteme
1.5.	Natürlichsprachliche Interfaces für den Dialog
2.	*Bildverarbeitungssysteme und Computergrafik*
2.1.	Digitale Bildverarbeitungssysteme
2.2.	Angewandte/spezialisierte Bildverarbeitungs- und Analysesysteme
2.3.	Computergrafik
3.	*CAD von Computersystemen*
3.1.	Beschreibungs- und Modellierungssysteme auf Architekturniveau
3.2.	CAD für VLSI- und ULSI-IC
3.3.	CAE für Computersysteme
4.	*Computernetze*
4.1.	Lokale Computernetze
4.2.	Hardware und Software für Netzprozessoren
4.3.	Intelligente Terminalnetze
4.4.	Verteilte Datenverarbeitung
5.	*Personalcomputer (PC)*
5.1.	Architekturfamilien neuer Generationen von PC
5.2.	Forschung zu PC für KI-Problemlösung
5.3.	Mensch-Maschine-Interfaceentwicklungen
6.	*Fehlertolerante Computersysteme*
6.1.	Automatische Hardwarediagnose
6.2.	Prinzipien der Interfaceorganisation von fehlertoleranten Systemen
6.3.	Architektur und Sicherungsmethoden für Fehlertoleranz
6.4.	Algorithmen und Programm für zuverlässige Computersysteme
6.5.	Zuverlässige Computersystemarchitektur zur Implementierung in Wafer-Scale-Integration (WSI)
7.	*Neue Massenspeicher*
7.1.	Massenspeicher auf der Basis optischer Aufzeichnungen und elektrischer Verstärkung, integrierte optische Massenspeicher; hierarchische holographische Massenspeicher; assoziative Massenspeicher
7.2.	Mehrebenen Silicium-Massenspeicher
7.3.	Zylindrische Magnetblasen-Massenspeicher
7.4.	Geräte zur hochdichten Digitaldatenaufzeichnung
7.5.	Datenschutz in Massenspeichern
8.	*Softwaretechnologie*
8.1.	Experimentalsystem für den automatischen Entwurf, die Produktion und Abarbeitung von Software für professionelle PC der neuen Computergeneration
8.2.	Entwicklung von Methoden zur Steigerung der Softwarezuverlässigkeit
8.3.	Softwareentwurfstechnologie und Systemarchitektur
9.	*Neue Informationsverarbeitungsalgorithmen und Systemarchitekturen*
9.1.	Entwicklung und Verfeinerung der Methoden und der Technologie der mathematischen Modellierung und der Rechenexperimente
9.2.	Entwicklung von Programmierhilfsmethoden für die Analyse und Filterung von Prozessen in stochastischen Systemen
9.3.	Hochleistungssysteme und Datenfluß-Computerarchitekturen
9.4.	Algorithmen für Parallelverarbeitung
10.	*Informatik im Bildungswesen*
10.1.	Erstellung von Modellen von Unterrichtsprozessen
10.2.	Erforschung und Entwicklung der Architektur von Lehr-Trainingskomplexen
10.3.	Aufbau von Experimentalzentren für Informatik im Unterricht

Bei der klassischen Abarbeitungsstrategie wird nun stets die Symmetrieregel aufgerufen und $R(c, a)$ nie festgestellt. Ein System mit einer wesentlich flexibleren Strategie des Problemlösers wurde z. B. in Kiew mit APROS (Adaptive Problem Solver) entwickelt. Ein noch offenes Problem bei der Entwicklung von PROLOG-Systemen, die Metawissen verwenden, besteht darin, daß diese Systeme eine Art „technisches", universelles Metawissen (z. B. über Strategien) benötigen. Andererseits wird auch problemabhängiges Metawissen benötigt, das entscheidend vom Anwendungsbereich abhängt. So wird man für den Beweis von Theoremen anderes Metawissen benötigen als für die Routinediagnose von bestimmten Krankheitsklassen. Andererseits wird dieses Metawissen nicht immer a priori „eingegeben" werden können, es muß zum größten Teil erlernt werden. Dazu werden Methoden für den Erwerb von Metawissen benötigt.

3. Expertensysteme und Inferenzprozesse

Eines der langfristigen Ziele der KI besteht in der Entwicklung von wissensverarbeitenden Systemen (KIPS). Einen wesentlichen Schritt in diese Richtung stellen Expertensysteme (ES) dar. Heute sind weltweit über 5000 Expertensysteme bekannt, von denen aber nur wenige zum Routineeinsatz gekommen sind. Die Wissensrepräsentation erfolgt meist durch die Anwendung von Produktionsregeln, semantischen Netzen, Frames und logischen Darstellungen [23] und [31].

ES haben aber als weitere wesentliche Komponenten eine bestimmte Art logisch zu schließen (Inferenzkomponente). Meist ist die Inferenzstrategie (z. B. durch Regelverkettung) fixiert und rein deduktiv (Expertensysteme 1. Generation). „On the other hand most of these expert systems provide only a limited depth being rather shallow systems, which were capable of slightly more than an intelligent retrieval", mußte T. Vamos nach langjähriger Arbeit bei der Entwicklung von medizinischen ES kürzlich einschätzen [24]. Induktive, analoge Schlußweisen und Schließen auf Metaebene fehlen bei fast allen heutigen Expertensystemen. Den menschlichen Experten zeichnet aber gerade die Fähigkeit aus, induktiv und analog zu schließen, belehrt werden zu können (externe Lernfähigkeit) und aus der eigenen Erfahrung beim Lösen früherer Probleme zu lernen (interne Lernfähigkeit). Daher besteht ein naheliegendes Ziel darin, Expertensysteme der 2. Generation zu entwickeln, die über folgende Merkmale verfügen:

1. Hierarchisch aufgebaute Wissensbasis (d. h. auch Zulassung von Metawissen und Tiefenwissen)
2. Flexible deduktive Inferenzkomponente (d. h. mehrere deduktive Inferenzstrategien stehen zur Auswahl, die problemangepaßt ausgewählt werden)
3. Wahlweise deduktive, induktive oder analoge Inferenz oder Metainferenz
4. Natürlichsprachiges Interface auf Fachsprachenebene
5. Erklärungskomponente, die nicht nur die Abarbeitung einer benutzten Strategie, sondern auch die Wahl der Strategie erklärt.

Als besondere Schwierigkeit hat sich die Formalisierung analoger Schlußweisen erwiesen [25] und [26]. Es müssen dazu Algorithmen entwickelt werden, die die Analogie von Prozessen entscheiden [27] und [28] (z. B. die Analogie chemischer Reaktionsgleichungen) bzw. die zu einem gegebenen Programm ein analoges Programm erzeugen (Programmtransformation durch Analogie). Der automatischen Synthese von PROLOG-Programmen durch Analogiealgorithmen sind auch in Japan gezielte Forschungsarbeiten gewidmet [29].

Aufgrund des Gesagten können folgende Voraussetzungen formuliert werden, die erfüllt sein müssen, damit Expertensysteme für den Routineeinsatz geeignet sind:

1. Die bisherigen Methoden der Befragung von Experten und des Zwängens von Expertenwissen in Regelform („spanische Stiefel") und die Eingabe in den Rechner sind zu ineffektiv (sog. Feigenbaumscher Flaschenhals). Vielmehr müssen Verfahren entwickelt werden, die das ES das Wissen verschiedener Wissensbereiche „lehren".
2. Das Expertensystem muß über eine flexible Abarbeitungsstrategie verfügen (deduktive und induktive Inferenzstrategien).
3. Interne Lernfähigkeit: Das ES muß über die Fähigkeit verfügen, aus eigener Erfahrung zu lernen, z. B. analog schließen zu können.
4. Für die Arbeit von ES müssen billige und schnelle Spezialprozessoren zur Verfügung stehen (32-bit-LISP- oder parallele PROLOG-Spezialprozessoren mit etwa 300—1000 KLIPS, bis zu 10 Mbyte interne und bis zu 600 Mbyte externe Speicherkapazität).

Nur bei einer erfolgreichen Lösung der genannten Probleme der KI-Forschung ist diese in der Lage, einen angemessenen Beitrag zur Entwicklung der 5. Rechnergeneration zu leisten und den Übergang von der Daten- zur Wissensverarbeitung zu ermöglichen.

Literatur

[1] Frege, G.: Begriffsschrift, eine der arithmetischen nachgebildete Formelsprache des reinen Denkens. Halle: Verlag L. Nebert 1879. — Gekürzter Nachdruck in: Berka, K.; Kreiser, L.: Logik-Texte. Berlin: Akademie-Verlag 1971.

[2] Moto-Oka, T.: Fifth Generation Computer Systems. Proc. Int. Conf. FGCS, Tokyo/Japan, Oct. 19—22, 1981. Amsterdam/New York: North Holland Publ. Comp. 1982. FGCS '84, Nov. 6—9, 1984. Ed. by ICOT, Tokyo 1984.

[3] Kowalski, R. A.: Predicate Logic as Programming Language. Proc. IFIP 1974. Amsterdam: North Holl. Publ. Comp. 1974.

[4] Kempe, V.: Denkzeuge der Zukunft. Spectrum 16 (1985) 8, S. 1—4.

[5] Giloi, W. K.: Die Entwicklung der Rechnerarchitektur von der von-Neumann-Maschine bis zu den Rechnern der „fünften Generation". Elektronische Rechenanlagen 26 (1981) 2, S. 55—70.

[6] Expert Systems User 1 (1985) 4, p. 7.

[7] Boehm, B. W.: The hardware/software cost ratio: Is it a myth? Computer, (1983) March pp. 78—80. — Boehm, B. W.: Software Engineering Economics, Englewood Cliffs/USA: Prentice-Hall 1981.

[8] Clocksin, W. F.; Mellish, C. S.: Programming in PROLOG. Berlin/Heidelberg/New York: Springer-Verlag 1981.

[9] Hier sei generell auf die Zeitschriften: „Future Generation Computing Systems" (FGCS). Amsterdam/New York: North Holland Publ. Comp. und „New Generation Computing" (NGC). Berlin (West)/ Tokyo: Ohmsha Ltd., beide 1982ff. verwiesen.

[10] Steffik, M.: Strategic Computing at DARPA: Overview and assessment. Comm. of ACM 28 (1985) 7, pp. 690—704.

[11] Proc. of Logic Programming Conf. '86, '87, '88. ICOT, 1986, 1987, 1988.

[12] Shapiro, E. Y.: A subset of concurrent PROLOG and its interpreter. ICOT Report TR-003, Tokyo 1983.

[13] Shapiro, E. Y.: Concurrent PROLOG. Collected papers. Vol. 1, 2. Cambridge, Mass./London: The MIT Press 1987.

[14] Ueda, K.: On the operational semantics of guarded Horn clauses. ICOT Techn. Memorandum. Tokyo, 1986.

[15] Sato, S.; Masakatsu, S.: Artificial Intelligence. Fujitsu Sci. Techn. J. 22 (1986) 3, pp. 139—181.

[16] Amundsen, M. et al.: Compact LISP machine. Texas Instruments Eng. J. (1986) 1/2, pp. 116—121.

[17] Chikayama, T.: ESP-Extended Self-Contained PROLOG — as a preliminary kernel language of Fifth Gen. Comp. New Generation Computing 1 (1983) 3, pp. 11—24.

[18] Futo, I.: Combined discrete/continuous modeling and problem solving. SCS Multiconference on AI, Graphics and Simulation. San Diego, California, 24—26 January 1986.

[19] Goldberg, A.: SMALLTALK-80. The interactive programming environment. Reading, Mass./USA: Addison-Wesley Publ. Corp. 1984.

[20] Mohr, I.: Common Talk. Eine objektorientierte Erweiterung von Common LISP. INFO '88, Dresden 1988. Band 1. S. 176ff.

[21] Dahn, I.: FLEXLOG. INFO '88, Dresden 1988. Band 1. S. 182ff.

[22] Harmelen, F. v.: A categorisation of meta-level architectures. AISB-87 Conference, April 1987.

[23] Frost, R.: Introduction to knowledge base systems. London: Collins Prof. and Techn. Books 1986.

(Fortsetzung auf S. 198)

W. Uebel, A. Friedrich, H. Hoffmann

Bestimmung der Farbanteile mehrfarbiger Druckvorlagen mit bildanalytischen Methoden

0. Einleitung

Eine Automatisierung des Mehrfarbendruckprozesses wird in erster Linie unter den Aspekten der Qualitätserhöhung und Qualitätseinhaltung während des Druckprozesses und im Anfahrprozeß gefordert. Damit ist eine erhebliche Materialeinsparung verbunden, da bei der hohen Bogengeschwindigkeit von 5 ... 10 m/s eine Off-line-Qualitätskontrolle stets große Ausschußquoten bedingt. Eine derartige Automatisierung verlangt einerseits eine schnelle Feststellung von relativ geringen Farbdruckfehlern und andererseits eine detaillierte Kenntnis über die quantitative und temporäre Wirkung der Veränderung der Druckparameter, um Qualitätsverschlechterungen frühzeitig zu erkennen und erfolgreich beeinflussen zu können. Die wesentlichsten Qualitätskontrollen sind im Hinblick auf Farbveränderungen, Tonen, Passerfehler, Folgen von Verschmutzungen u. ä. erforderlich. Die Automatisierung zwingt dabei zur Echtzeitmessung im Druckbild während des Druckprozesses ohne Entnahme eines Kontrollbogens. Neben der Feuchtmittelregulierung stellt die Farbgebungssteuerung die wesentlichste Beeinflussung dar. Sie setzt die Ermittlung der Farbanteile in der Druckvorlage voraus, wobei die Messung direkt im Druckbild und nicht an einem seitlich angebrachten Kontrollstreifen erfolgen muß, um die tatsächlichen Farbanteile im Zusammendruck der Farbrasterpunkte in den einzelnen Zonen des Druckbildes zu erfassen.

Dr. rer. nat. WOLFGANG UEBEL (46) studierte von 1962 bis 1967 Physik an der Humboldt-Universität zu Berlin. 1971 Promotion A. Seitdem wissenschaftlicher Mitarbeiter am Zentralinstitut für Kybernetik und Informationsprozesse der AdW der DDR.

Dipl.-Ing. ANDREA FRIEDRICH (31) studierte von 1976 bis 1981 Informationsverarbeitung an der Technischen Universität Dresden. Seit 1981 wissenschaftliche Mitarbeiterin am Zentralinstitut für Kybernetik und Informationsprozesse der AdW der DDR.

Dipl.-Ing. HERIBERT HOFFMANN (32) studierte von 1977 bis 1982 Gerätetechnik an der Technischen Hochschule Ilmenau. Danach Hardwareentwickler im VEB Studiotechnik Berlin auf dem Gebiet der CCD-Technik und Bildvorverarbeitung. Seit 1987 wissenschaftlicher Mitarbeiter am Zentralinstitut für Kybernetik und Informationsprozesse der AdW der DDR.

Der Aufbau eines geeigneten Gerätesystems setzt die Schaffung der verfahrensmäßigen und algorithmischen Grundlagen der Farbanteilbestimmung voraus. Dies bildet den wesentlichen Inhalt der dargestellten Ergebnisse. Da die Beurteilung am fertigen Druckbild erfolgt, stellt die automatische Bildanalyse eine geeignete methodologische Grundlage der automatisierten Qualitätssicherung im Druckprozeß dar.

1. Problembeschreibung und Druckbildmodell

Die Signale zur qualitätseinhaltenden Farbgebungssteuerung lassen sich aus dem Verhältnis der Farbanteile der Primärdruckfarben Cyan (c), Gelb (y), Magenta (m) und Schwarz (s) zu den Sollwerten eines als partikuläres Normal eingestuften Vergleichsdrucks ableiten. Die Ermittlung der Farbanteile am fertigen Druckerzeugnis unmittelbar nach dem letzten Druckvorgang bildet somit die Grundlage für die gewünschte Qualitätskontrolle.

Zur Lösung der Aufgabe wird folgendes Druckbildmodell zugrunde gelegt:

Der Rastervierfarbendruck besteht aus Farbrasterpunkten, deren Fläche bei einem Flächenbedeckungsgrad von $<50\%$ als kreisförmig angenommen wird (Bild 1). Die Mittelpunkte der Farbrasterpunkte R_{ij} sind identisch mit den Mittelpunkten der Rasterquadrate (x_i, y_j) eines Rechteckrasters mit äquidistanten Rasterabständen, d. h.

$$x_i = i \cdot x_0 , \quad i = 0, \dots , m , \qquad y_j = j \cdot y_0 , \quad j = 0, \quad , n . \tag{1}$$

x_0 und y_0 sind durch das Druckverfahren gegeben. In der Regel gilt $x_0 = y_0$ mit $100\ \mu m \leqq x_0 \leqq 300\ \mu m$. Diese Quadratraster der vier Primärfarben sind um feste Winkel gegeneinander verdreht. Die Winkel haben z. B. die Werte

$$\varphi_c = 15° \quad \varphi_m = 75° , \qquad \varphi_y = 0° , \quad \varphi_s = 45°$$

mit einer Genauigkeit von $\pm 1°$. In der Regel weisen die Quadratraster der Primärfarben untereinander einen zufälligen Versatz aufgrund von Passerfehlern auf. Mehrere Quadratraster (>100) bilden ein Meßsegment. Die Farbanteilbestimmung wird nur in

(Fortsetzung von S. 197)

[24] VAMOS, T. et al.: Pattern recognition combined with logic — a decade's experience with an unusual medical expert system. Working paper E/39, SZTAKI, Budapest 1986.

[25] Analogical and Inductive Inference. Int. Workshop, October 6—10, 1986. In: Lecture Notes in Computer Science. Vol. 265. Berlin/Heidelberg/New York: Springer-Verlag 1987.

[26] PÖTSCHKE, D.: Bibliography on analogical reasoning. (To be published in: J. of Symbolic Computation).

[27] PÖTSCHKE, D.: Toward a mathematical theory of analogical reasoning. Proc. 1982 European Conf. on Artificial Intelligence. Orsay/France, July 11—14, 1982.

[28] PÖTSCHKE, D.: Analogical reasoning using graph transformations. In: Lecture Notes in Computer Science. Vol. 265. Berlin/Heidelberg/New York: Springer-Verlag 1987.

[29] YOKOMORI, T.: Logic Programm Forms. New Generation Computing 4 (1986), pp. 305—319.

[30] PLANDER, I.. Projects of the new Generation Computer Systems and Informatics. Computers and Art. Int. 4 (1985) 6, pp. 551—565.

[31] PÖTSCHKE, D.; LUNZE, J.: Künstliche Intelligenz und Automatisierungstechnik. Berlin: VEB Verlag Technik (erscheint 1989).

msr 8954

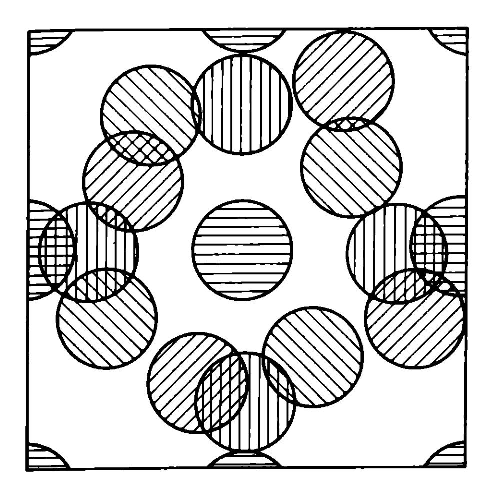


Bild 1.
Rastervierfarbendruck bei einem Flächenbedeckungsgrad von <30%
≡ gelb;
||| schwarz;
/// cyan;
\\\ magenta

ausgewählten Meßsegmenten vorgenommen, deren Auswahlbedingungen noch festzulegen sind (s. Abschn. 2.3.). Das Meßsegment wird bei der bildanalytischen Verarbeitung durch das Bildraster in quadratische Elemente (Pixel) unterteilt. Liefert die Aufnahmekamera kein quadratisches Bildraster, so wird durch geeignete Maßnahmen (s. Abschn. 3.) das Bildraster quadratisch gemacht. Der absorbierte Strahlenfluß ist proportional zur Farbmenge, d. h. sowohl von der Fläche F der Farbrasterpunkte als auch von der Dickenverteilung $d(x, y)$ abhängig.
Unter der Annahme einer konstanten Farbschichtdicke wird die Bestimmung der Farbanteile zunächst auf eine Bestimmung der Flächenbedeckung der einzelnen Farben zurückgeführt. Auf die Bestimmung der Farbschichtdicke wird am Ende des Abschn. 2.2. eingegangen.
Ursachen von Farbabweichungen können einmal drucktechnischer Natur sein, wie Farbdichteänderungen, Farbmengenänderungen, falsche Feuchtmittelführung, falsche Druckfarbenreihenfolge u. a.
Die wesentlichen Ursachen liegen jedoch nach [1] in Passerabweichungen, bedingt durch Ungenauigkeiten der Druckmaschine bzw. Dehnungen des Bedruckstoffes und in der Moiré-Bildung, die aber durch die Winkelung der Primärfarben auf ein nicht wahrnehmbares Maß reduziert wird.

2. Bestimmung des Flächenbedeckungsgrades der Primärfarben

Nach dem Grassmannschen Gesetz [2] wird jede Farbempfindung durch drei Bestimmungsgrößen eindeutig beschrieben. Sie bilden die Farbvalenz

$$X = \frac{1}{k} \sum_\lambda S_\lambda \beta_\lambda \bar{x}_\lambda , \qquad (2)$$

$$Y = \frac{1}{k} \sum_\lambda S_\lambda \beta_\lambda \bar{y}_\lambda ,$$

$$Z = \frac{1}{k} \sum_\lambda S_\lambda \beta_\lambda \bar{z}_\lambda ,$$

$$\frac{1}{k} = \frac{100}{\sum_\lambda S_\lambda \bar{y}_\lambda},$$

wobei S_λ die spektrale Verteilung der Lichtquelle, β_λ die Remissionsfunktion der Körperfarbe und $\bar{x}_\lambda$, $\bar{y}_\lambda$, $\bar{z}_\lambda$ die Normspektralwertkurven sind. Diese können dabei durch beliebige Spektralwertkurven eines anderen Primärvalenzsystems ersetzt werden. Beim Vierfarbendruck gibt es durch Bildung der Sekundär-, Tertiär- und Quartiärfarben 16 Farbreize, denen 16 Farbvalenzen (X_i, Y_i, Z_i) entsprechen. Diese 16 Farbreize addieren sich im Auge zu einer Farbvalenz (X, Y, Z). Die Farbreize der Teilfarben sind proportional ihrem Flächenanteil ψ_i an der Farbempfindung beteiligt, so daß nach [3] bis [5] gilt:

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = \psi_0 \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} + \psi_1 \begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \end{pmatrix} + \quad + \psi_4 \begin{pmatrix} X_4 \\ Y_4 \\ Z_4 \end{pmatrix}$$
$$+ \psi_{12} \begin{pmatrix} X_{12} \\ Y_{12} \\ Z_{12} \end{pmatrix} + \psi_{13} \begin{pmatrix} X_{13} \\ Y_{13} \\ Z_{13} \end{pmatrix} + \quad + \psi_{34} \begin{pmatrix} X_{34} \\ Y_{34} \\ Z_{34} \end{pmatrix}$$
$$+ \psi_{123} \begin{pmatrix} X_{123} \\ Y_{123} \\ Z_{123} \end{pmatrix} + \psi_{124} \begin{pmatrix} X_{124} \\ Y_{124} \\ Z_{124} \end{pmatrix} + \ldots + \psi_{234} \begin{pmatrix} X_{234} \\ Y_{234} \\ Z_{234} \end{pmatrix}$$
$$+ \psi_{1234} \begin{pmatrix} X_{1234} \\ Y_{1234} \\ Z_{1234} \end{pmatrix}. \qquad (3)$$

Da die Lage der Farbrasterpunkte zueinander infolge der Passerschwankungen zufallsbedingt ist, lassen sich die ψ_i nach [3] und [4] aufgrund eines Wahrscheinlichkeitsansatzes berechnen. Das Ergebnis ist die Neugebauer-Gleichung

$$\begin{pmatrix} X \\ Y \\ Z \end{pmatrix} = (1-\alpha_1)(1-\alpha_2)(1-\alpha_3)(1-\alpha_4) \begin{pmatrix} X_0 \\ Y_0 \\ Z_0 \end{pmatrix} +$$
$$+ \alpha_1(1-\alpha_2)(1-\alpha_3)(1-\alpha_4) \begin{pmatrix} X_1 \\ Y_1 \\ Z_1 \end{pmatrix} +$$
$$+ (1-\alpha_1)\alpha_2(1-\alpha_3)(1-\alpha_4) \begin{pmatrix} X_2 \\ Y_2 \\ Z_2 \end{pmatrix} +$$
$$\ldots + \alpha_1\alpha_2\alpha_3\alpha_4 \begin{pmatrix} X_{1234} \\ Y_{1234} \\ Z_{1234} \end{pmatrix}, \qquad (4)$$

wobei die α_1, α_2, α_3, α_4 die Flächenbedeckungsgrade der 4 Primärfarben sind. Dieses System aus 3 Gleichungen mit 4 Unbekannten ist nicht lösbar. Die Hinzunahme einer 4. Spektralwertkurve bringt keine Lösung, da die Farbe Schwarz im gesamten sichtbaren Spektralgebiet eine gleichverteilte niedrige Remission aufweist. Jeder Beitrag zu Schwarz kann auch ein Beitrag zu den 3 Buntfarben sein und umgekehrt. Das heißt, daß die Schätzung der Farbanteile allein durch spektrale Messungen im Meßsegment unmöglich ist. Auch im Infrarotbereich zeigen nur spezielle schwarze Farben eine signifikante Absorption.

2.1. Lösungsweg

Unter diesen Bedingungen wird als Lösungsweg die separate Bestimmung des Flächenanteils α_4 der Farbe Schwarz durch bildanalytische Methoden und damit Eliminierung der 4. Unbekannten α_4 in der Neugebauer-Gleichung angesehen. Zur bildanalytischen Bestimmung werden zur Bilderzeugung das gleiche Meßsegment, in dem die Farbvalenzen (X_i, Y_i, Z_i) gemessen werden, mittels einer CCD-Flächenkamera in 3 Spektralbereichen digital aufgenommen und aus den 3 Bildern der gesuchte Flächenanteil α_4 ermittelt. Eine Aussage über die Dicke der schwarzen Farbrasterpunkte ist nicht erforderlich, da Schwarz auch bei der geringstmöglichen Dicke das einfallende Licht fast vollständig absorbiert [1].

2.2. Bildanalytische Bestimmung des Flächenbedeckungsgrades der Farbe Schwarz

Die Abtastung der 3 Farbauszugsbilder in den Spektralbereichen Rot, Grün und Blau mittels einer CCD-Flächenkamera erfolgt mit einer Auflösung von 18 µm/Pixel. Das Bildformat beträgt 128 × 128 Pixel, so daß ein Meßsegment von 2,3 × 2,3 mm² erfaßt wird. Die Analogsignale werden in 64stufige Grauwerte (6 bit) gewandelt. Die Beleuchtung erfolgt mit Licht der Normlichtart C oder einer ähnlichen Charakteristik. Geeignet ist eine Xenonblitzbeleuchtung.

Die Filterung muß in den für die 3 Buntfarben typischen spektralen Absorptionsbereichen erfolgen. Im Idealfall entspricht die Transmissionskurve des Filters der Absorptionskurve der jeweiligen Buntfarbe. Jedoch sind ebenso schmalbandige Filter in für die jeweilige Absorption typischen Spektralbereichen geeignet, sofern die Filter eine ausreichende Durchlaßfähigkeit besitzen. Benutzt werden die Bandpaßfilter BPF 450/35c (blau), BPF 550/50c (grün) und BPF 625/60c (rot) vom Kombinat VEB Carl Zeiss JENA. Ihre Durchlaßfähigkeit ist $>70\%$.
Zur algorithmischen Beschreibung des Lösungsverfahrens seien zunächst einige Definitionen vorangestellt.
Es sei g_b die Menge aller Rasterpunkte des Blauauszuges. Den Mengenelementen pg_b, den Pixeln des Bildes, sind die Grauwerte zugeordnet:

$$g_b = [pg_b(i,j): i = 1, \dots, 128; j = 1, \dots, 128], \quad pg_b \in [0, 1, \dots, 63]. \tag{5}$$

Entsprechendes gilt für die Menge g_g der Rasterpunkte des Grünauszuges und für die Menge g_r des Rotauszuges mit entsprechenden Werten pg_g und pg_r. Beim Vierfarbendruck mit den Farben c, m, y und s besteht die Menge g_l ($l \in [b, g, r]$) aus 16 Untermengen der Rasterpunkte der 4 Primärfarben, der 6 Sekundärfarben, der 4 Tertiärfarben, der Quartiärfarbe sowie der Farbe w des Bedruckstoffes, d. h.

$$g_l = [[c], [y], [m], [s], [cy], [cm], [cs], [ym], [ys], [ms], [cmy], [cms], [yms], [cys], [cmys], [w]]. \tag{6}$$

Zum Beispiel ist $[c]$ die Menge aller derjenigen Pixel, die nur der Farbe Cyan entsprechen. Dabei werden Digitalisierungseffekte unberücksichtigt gelassen.
Bei der Binärisierung eines Grauwertbildes g_l mit einer Schwelle th_k entsteht die binäre Menge

$$b_l = [pb_l(i,j): i = 1, \dots, 128; j = 1, \dots, 128], \quad pb_l \in [0, 1], \quad l \in [b, g, r]. \tag{7}$$

Des weiteren werden alle Elemente mit $pb_l = 0$ zur Untermenge $[w]$ zusammengefaßt, so daß die übrigen 15 Untermengen nur noch Elemente $pb_l = 1$ enthalten. Je nach Wahl der Schwelle werden die Untermengen verschieden mächtig oder auch leer sein.
Schließlich sei folgende Verknüpfung definiert:

Die Verknüpfung $b_k \oplus b_l$ der binären Mengen b_k und b_l ergibt eine Menge b_v, deren Elemente $pb_v(i,j) = 1$ genau dann sind, wenn $pb_k(i,j) = 1$ und $pb_l(i,j) = 1$ gilt.

Zur Bestimmung des Flächenbedeckungsgrades der Farbe Schwarz werden zunächst die Schwerpunkte der schwarzen Farbrasterpunkte ermittelt und als Objektliste abgespeichert. Zu diesem Zweck sind die 3 Farbauszugsbilder so zu binärisieren, daß

$$b_b = [[y], [s*], [cmy], [cy], [my]], \quad b_g = [[m], [s*], [cmy], [cm], [my]], \quad b_r = [[c], [s*], [cmy], [cy], [cm]], \tag{8}$$

mit $[s*] = [[s], [cs], [ms], [ys], [cms], [cys], [mys], [cmys]]$ gilt. Das ist stets möglich, da die Transmissionscharakteristiken der verwendeten Filter den Absorptionscharakteristiken der zugehörigen komplementären Druckfarben entsprechen. Die Bestimmung der geeigneten Binärisierungsschwellen erfolgt aus den Histogrammen, in dem die Lage des zur Farbe Schwarz gehörigen Maximums als Schwelle thschwarz verwendet wird (Bild 2). Die Verknüpfung

$$b_s = b_b \oplus b_g \oplus b_r \tag{9}$$

ergibt eine Menge, die nur noch aus den Untermengen $b_s = [[s*], [cmy]]$ besteht. b_s enthält Binärobjekte, deren Schwerpunkte den Schwerpunkten der schwarzen Farbrasterpunkte entsprechen. Bild 3 zeigt als Beispiel die 3 Farbauszugsbilder g_b, g_g, g_r sowie das Verknüpfungsergebnis b_s. Die Binärobjekte werden mit Hilfe eines Objektisolationsverfahrens (Zeilenkonzidenzverfahren) ermittelt, in dem während der Markierung der Einzelobjekte im Bild deren Koordinaten des umschreibenden Rechtecks *imin,imax,jmin,jmax* und deren Fläche f mit berechnet werden. Die Koordinaten des umschreibenden Rechtecks gestatten eine Schätzung des Schwerpunktes mit ausreichender Genauigkeit nach

$$is = (imax - imin)/2, \quad js = (jmax - jmin)/2. \tag{10}$$

Mit den Merkmalen *imin,imax,jmin,jmax,f* werden die Einzelobjekte nach Fläche und Lage klassifiziert, so daß nur noch Objekte mit $f \geqq 2$ und einem Randabstand von mindestens 3 Pixel in einer Objektliste abgespeichert werden.
Dieser Algorithmus zur Schwerpunktsermittlung ist nicht zur Bestimmung des gesuchten Flächenbedeckungsgrades durch Verwendung höherer Schwellen geeignet, da mit Erhöhung der Binärisierungsschwelle *thschwarz* weitere Untermengen zu b_l ($l = b, g, r$) hinzukommen, die bei der Verknüpfung zu b_s nicht beseitigt werden. Insbesondere liefern die Überdeckungen der Farben Cyan und Magenta sowie Cyan und Gelb abweichend vom Idealfall Grauwerte im Bereich der Farbe Schwarz.
Aus diesen Gründen ist ein Algorithmus entwickelt worden, der in Anlehnung an die Algorithmen zum Gebietswachstum (region

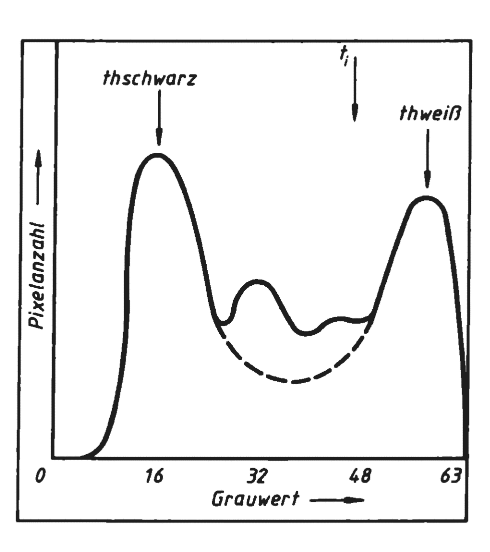


Bild 2. Typisches Grauwerthistogramm eines Farbauszugbildes gestrichelt bei Vorlage schwarzer Farbrasterpunkte; beispielhaft sind die Schwellen *thschwarz*, *thweiß* (Grauwertmaximum des Bedruckstoffes) und t_i, $i \in [b, g, r]$, angegeben

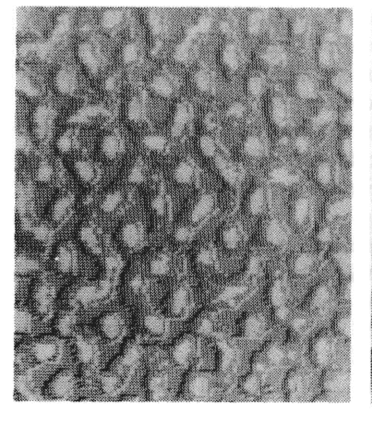
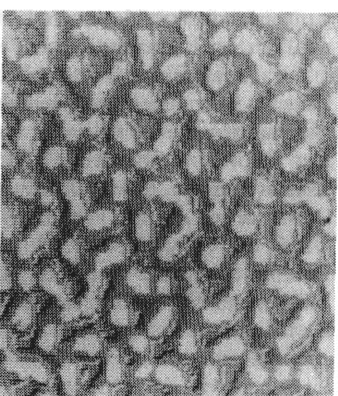
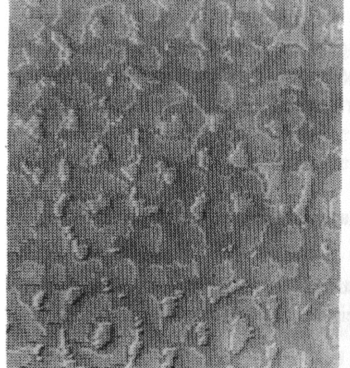
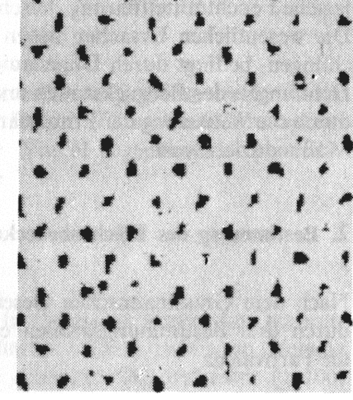





Bild 3. Farbauszugsbilder g_b (l. u.), g_g (r. o.) und g_r (l. o.) eines Druckes mit Flächenbedeckungsgraden von 30% Cyan, 30% Magenta, 30% Gelb und 30% Schwarz sowie das Verarbeitungsresultat b_s (r. u.)

growing) als „bedingtes Gebietswachstum“ bezeichnet werden soll. Ausgangspunkt sind die in der Objektliste vorliegenden Schwerpunkte der schwarzen Farbrasterpunkte. In konzentrischen Pixelringen um den Schwerpunkt (Bild 4) werden alle Pixel in den 3 Farbauszugsbildern g_b, g_g und g_r auf die Erfüllung der Bedingungen

$$\begin{aligned} pg_b(i,j) &\leqq t_b\,, \\ pg_g(i,j) &\leqq t_g\,, \\ pg_r(i,j) &\leqq t_r \end{aligned} \qquad (11)$$

überprüft. Die Schwellen t_b, t_g und t_r kennzeichnen den Grauwertübergang zwischen dem schwarzen Farbrasterpunkt und dem Bedruckstoff. Der Algorithmus prüft sequentiell die Pixel vom inneren zum äußeren Ring und zählt alle Pixel bis zur erstmaligen Verletzung einer der obigen Bedingungen. Alle Pixel des Ringes, in dem der Abbruch erfolgt, die die obigen Bedingungen noch erfüllen, werden mitgezählt. Diese Anzahl repräsentiert die gesuchte Fläche, aus der der Flächenbedeckungsgrad in % aufgrund der bekannten Geometrie der Farbrasterpunktanordnung berechnet wird. Der Wachstumsprozeß wird somit nicht bis zur vollständigen Segmentierung des Bildes fortgesetzt, sondern bricht aufgrund der Bedingungen vorher ab. Die Anwendung erfolgt auf alle Schwerpunkte der Objektliste. Aus den einzelnen Flächen (in der benutzten Auflösung sind es 100 ... 120 Objekte) wird unter Beseitigung extremer Differenzen der Mittelwert berechnet. Die Schwellen t_b, t_g und t_r sind aus den jeweiligen Histogrammen ableitbar. Sie kennzeichnen den Wendepunkt (Nulldurchgang der 2. Ableitung) in der aufsteigenden Flanke des zum Bedruckstoff gehörenden Histogrammpeaks (s. Bild 2). Das beschriebene Verfahren macht folgende Voraussetzungen an die Auswahl des Meßsegmentes:

- Die schwarzen Farbrasterpunkte müssen isoliert sein und mindestens an einer Stelle eine gemeinsame Kante mit dem Bedruckstoff haben. Diese Voraussetzung ist erfüllt, wenn der Flächenbedeckungsgrad < 50 % ist und im Rundraster gedruckt wird.
- Innerhalb des Meßsegmentes wird Farbhomogenität angenommen, da über alle Flächenwerte der Farbrasterpunkte gemittelt wird.

Da drucktechnisch bedingte Über- und Unterfärbungen sowohl die Dicke als auch den Flächenbedeckungsgrad beeinflussen, ist für eine genauere Farbanteilbestimmung auch eine Schätzung der Farbschichtdicke notwendig. Der bildanalytische Zugang gestattet eine solche Dickenschätzung der Buntfarben über den mittleren Grauwert der Pixel der Untermengen [*c*], [*m*] und [*y*]. Die Meßgenauigkeit wird vom Grauwertumfang der Pixel bestimmt. Durch logische Verknüpfungen geeignet binärisierter Bilder können die monochromatischen Pixelmengen [*c*], [*m*] und [*y*] ermittelt werden.

2.3. Experimentelle Resultate

Mit Hilfe des im Abschnitt 3. beschriebenen Experimentalsystems wurden am unbewegten Druckbogen die in der Tafel gezeigten Flächenbedeckungsgrade für verschiedene Überdeckungen ermittelt. Zum Vergleich sind die mikroskopisch ermittelten Werte gegenübergestellt. Die Ungenauigkeit der Reproduzierbarkeit des bildanalytisch berechneten Flächenbedeckungsgrades der Farbe Schwarz

				5	5	5				
			5	4	4	4	5			
		5	4	3	3	3	4	5		
	5	4	3	2	2	2	3	4	5	
5	4	3	2	1	1	1	2	3	4	5
5	4	3	2	1	S	1	2	3	4	5
5	4	3	2	1	1	1	2	3	4	5
	5	4	3	2	2	2	3	4	5	
		5	4	3	3	3	4	5		
			5	4	4	4	5			
				5	5	5				

Bild 4. **Pixelringe zum Algorithmus „bedingtes Gebietswachstum“**

konnte mit < 1 % bestimmt werden. Weitere Resultate werden in [10] dargestellt.

Tafel. Beispiele für bildanalytisch (bil) ermittelte und berechnete (ber) Flächenbedeckungsgrade der vier Primärfarben Cyan (c), Gelb (y), Magenta (m) und Schwarz (s) im Vergleich mit mikroskopisch (mik) bestimmten Werten

Druckbeispiel	Flächenbedeckungsgrad in %							
	Schwarz		Cyan		Gelb		Magenta	
	bil	mik	ber	mik	ber	mik	ber	mik
c			32	32				
y					40	40		
m							35	34
s	31	31						
y + m					35	37	33	32
y + s	31	30			45	43		
c + y + m			32	27	32	34	20	19
c + m + s	27	27	37	33			26	26
c + y + m + s	26	25	33	31	33	31	22	23

3. Hardware des Experimentalsystems

Das Bildaufnahmesystem besteht aus einem Aufnahmekopf und einem Videoeingabe- und -speichermodul in Form einer Erweiterungskarte für einen 16-bit-PC. Der Aufnahmekopf enthält die Beleuchtungseinrichtung, bestehend aus einem Xenonblitzgerät mit Strahlfokussierung, und Fotodioden zur Messung der drei Komponenten der Farbvalenz des beleuchteten Meßsegmentes sowie zur Erfassung der Beleuchtungsstärke am Meßobjekt als Referenzwert. Eine CCD-Flächenkamera mit Objektiv zur Aufnahme der drei digitalen Farbauszugsbilder ist fest mit dieser Einheit verbunden. Weiterhin befinden sich am Aufnahmekopf die nachgeschalteten Analogverstärker für die Fotodioden sowie eine galvanische Entkopplung zur Auslösung des Lichtblitzes durch den PC. Von der CCD-Kamera gelangt außer dem Videosignal noch der Bildpunkttakt zum Bildeingabe- und -speichermodul. In Verbindung mit zwei programmierbaren 4-bit-Vorzählern können durch den Bildpunkttakt in Horizontal- sowie durch die abgetrennten Horizontalsynchronimpulse in Vertikalrichtung die Anfangskoordinaten des aufzunehmenden Meßsegmentes festgelegt werden. Das Bildpunktmittenraster der verwendeten CCD-Matrix hat ein Seitenverhältnis von 5:3 in Horizontalrichtung. Um eine Annäherung an ein quadratisches Bildpunktraster zu erzielen, werden bei der Aufnahme in den Halbbildern Zeilen entsprechend so ausgeblendet, daß bei einer Vollbildwiedergabe jede dritte Zeile fehlt. Durch diese Korrektur erhält man ein hinreichendes Seitenverhältnis von 10:9. Das Videosignal wird mit dem Bildpunkttakt über einen 6-bit-AD-Wandler digitalisiert. Die Referenzspannung für die maximale Videoamplitude ist über einen 6-bit-DA-Wandler programmierbar. Der Bildspeicher hat eine Kapazität von 16 Kbyte, wobei für jeden Bildpunkt 1 byte reserviert ist. Um ein Einschreiben bei einer Bildpunktfrequenz von etwa 7,2 MHz zu ermöglichen, werden zwei aufeinanderfolgende Bildpunkte parallel abgespeichert. Durch Nutzung eines aus dem Synchrongemisch generierten Halbbildsignals werden bei einer Aufnahme zwei aufeinanderfolgende Halbbilder im Videospeicher als Vollbild verschachtelt abgelegt. Außer bei Bildaufnahmen ist der Bildspeicher direkt im Adreßraum der 8086-CPU zum Beschreiben oder Lesen ansprechbar.

Die Meßwerte der Komponenten der Farbvalenz und der Beleuchtungsstärke sind über Eingabeports mit jeweils 10 bit Auflösung vom PC aufrufbar.

Den zeitlichen Zusammenhang einiger wichtiger Komponenten des Experimentalsystems bei einer Bildaufnahme zeigt Bild 5.

Zusammenfassung

Automatische Qualitätskontrollen im Farbdruckprozeß zielen in erster Linie auf eine Farbgebungssteuerung ohne Entnahme eines

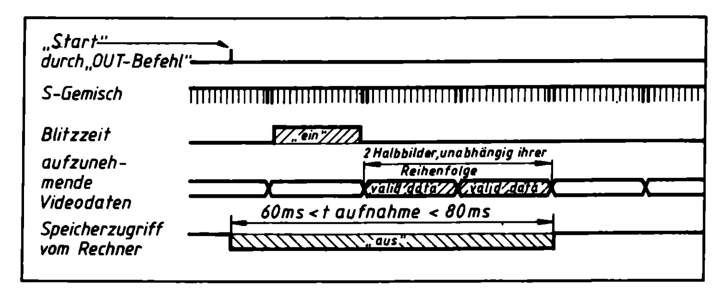


Bild 5. Zeitlicher Verlauf ausgewählter Signale während einer Bildaufnahme

Kontrollbogens. Die erforderlichen Steuersignale lassen sich aus dem Vergleich der Farbanteile der Primärdruckfarben Cyan, Gelb, Magenta und Schwarz zu den Sollwerten eines als Original eingestuften Vergleichdruckbogens ableiten. Auf der Grundlage des Neugebauer-Gleichungssystems können die Flächenbedeckungsgrade der Buntfarben durch fotoelektronische Messung der Farbvalenzen bestimmt werden. Da dieses Vorgehen für die Farbe Schwarz aufgrund der speziellen Remissionscharakteristik nicht anwendbar ist, wird deren Flächenbedeckungsgrad bildanalytisch ermittelt, indem vom Meßsegment mittels einer CCD-Flächenkamera 3 Farbauszugsbilder erstellt werden, die mit Bildanalysemethoden ausgewertet werden. Durch Blitzlichtbeleuchtung im Mikrosekundenbereich wird die Bildaufnahme am laufenden Druckbogen ermöglicht.

Die Autoren danken besonders Herrn Prof. Dr. sc. K. Fritzsch für die ständige Betreuung dieser Arbeiten sowie viele Hinweise und Diskussionen.

Literatur

[1] Häuser, F.: Die Entstehung des Farbeindrucks bei der autotypischen Farbmischung. Dissertation, Universität Bonn, Institut für Kartographie und Topographie 1982.

[2] Lang, H.: Farbmetrik und Farbfernsehen. München/Wien: R. Oldenbourg Verlag 1978.

[3] Neugebauer, H. E. J.: Die theoretischen Grundlagen des Mehrfarbendrucks. Zeitschrift f. wiss. Photographie, Photophysik und Photochemie 36 (1937) 4, S. 73—89.

[4] Neugebauer, H. E. J.: Theorie des Vierfarbendrucks mit einem Schwarzdruck. Zeitschrift f. wiss, Photographie, Photophysik und Photochemie 36 (1937) 8/9, S. 169–170.

[5] Scheuter, K. R.: Etwas zum Farbendruck. Der Druckspiegel (1979) 2, S. 180—193.

[6] Scheuter, K. R.: Grenze der Rastertechnik und Neubeginn (vom amplituden- zum frequenzmodulierten Bildaufbau). Der Druckspiegel DRUPA (1986), S. 490—497.

[7] Baufeldt, U.; Dorra, M.; Rösner, H.; Scheuermann, J.; Walk, H.: Informationen übertragen und drucken — Lehr- und Arbeitsbuch für das Berufsfeld Drucktechnik. Itzehoe: Verlag Beruf+Schule 1985.

[8] Richter, M.: Einführung in die Farbmetrik. Sammlung Göschende Gruyter Nr. 2608. Berlin: Walter de Gruyter 1976.

[9] Hoffmann, K.: Zusammenhang zwischen Extinktion bzw. Transmission und Remission Licht streuender Farbauflagen auf weißem Untergrund. Farbe und Lack 76 (1970) 7, S. 665—673.

[10] Friedrich, A.; Uebel, W.; Hoffmann, H.: Determination of colour dot ratio in multi-colour printing by means of picture analysis. CAIP '89, Leipzig, September 1989.

msr 9004

E. Bieber

Bildanalyse anhand der Dichte kritischer Punkte

0. Einleitung

Bei der Analyse von Bildern werden häufig Fragen nach bestimmten markanten Bildteilen formuliert. Derartige Punkte oder Gebiete können z. B. Fehlerstellen oder anders definierte Besonderheiten markieren. Es besteht oft die Notwendigkeit, diese lokalen Abweichungen von der globalen Bildstruktur zu extrahieren. Lokale Häufungen von derartigen Punkten sind ein Hinweis auf größere Abweichungen von der Normalität des Bildes.

Diese Probleme lassen sich anhand der Dichte kritischer Punkte lösen. Bei diesem Verfahren werden allgemeine Kriterien für die Abweichungen von der globalen Bildstruktur festgelegt. Diese Kriterien können in sich schon eine Bewertung enthalten oder über eine Bewertung der Eigenschaften einer Abweichungsrangordnung zugeführt werden.

1. Grundsätzliches Vorgehen bei der Analyse

Mit Hilfe der Anzahl von derart bewerteten Punkten je Flächeneinheit oder der Dichte kritischer Punkte kann die Analyse nach besonders stark markierten Gebieten vereinfacht werden. So lassen sich z. B. Gebiete extrahieren, bei denen die Dichte einen kritischen Wert (kritische Dichte) überschreitet. Ebenso lassen sich lokale Maxima in einem „Dichtegebirge" bestimmen.

Besonderes Augenmerk ist bei der Analyse auf die Auswahl adäquater Kriterien und eine sinnvolle, dem Analyseziel angepaßte Bewertung zu legen. Die Schätzung der Dichte kann als Summation der bewerteten Punkte in einem über das Bild gleitenden Fenster (z. B. quadratisch oder kreisförmig) ausgeführt werden. Die Analyse besteht dann im Endeffekt in der Auswertung des erzielten Dichtebildes („Dichtegebirge").

Etappen der Analyse

1. Vorbereitung des Quellbildes (allgemeine Bildverbesserung)
2. Definition von Bewertungskriterien
3. Bewertung der Parameter
4. Dichteschätzung
5. Auswertung des Dichtebildes

2. Anwendung der Analyse

In der konkreten Anwendung wurde diese Analysemethode bei der Suche nach Fehlerstellen in Reifenhologrammen benutzt. Dazu werden zwei Verschiebungszustände der Reifen bei verschiedenem Außendruck in ein Hologramm gebracht. Die entstehenden Interferenzen geben Auskunft über Deformationen. An den vermeintlichen Fehlerstellen (Ablösung von Gummi und Gewebe, Geweberisse, Blasen usw.) treten deutliche Deformationen hervor, was sich in der lokalen Häufung und starken Krümmung von Interferenzstreifen zeigt.

Zur automatischen Fehlererkennung und -bewertung der Reifen ist die Vorverarbeitung der Quellbilder erforderlich (Akkumulation, Filterung, Helligkeitsausgleich, Skelettierung usw). Ziel der Vorverarbeitung ist das Skelett der Interferenzstreifen.

Als adäquates Kriterium für die Bewertung wurde für die Skelette die Krümmung der Skelettlinien gewählt. An den Fehlerstellen tre-

Dr.-Ing. Erdmann Bieber (34) studierte von 1975 bis 1979 an der Sektion Informationstechnik der TU Dresden. Anschließend dort wissenschaftlicher Assistent. 1984 Promotion A auf dem Gebiet „Technische Bauakustik" Von 1983 bis 1986 wissenschaftlicher Mitarbeiter an der Bauakademie Berlin. Seit 1987 wissenschaftlicher Mitarbeiter im Zentralinstitut für Kybernetik und Informationsprozesse (ZKI) der AdW der DDR, Bereich Bildverarbeitung.

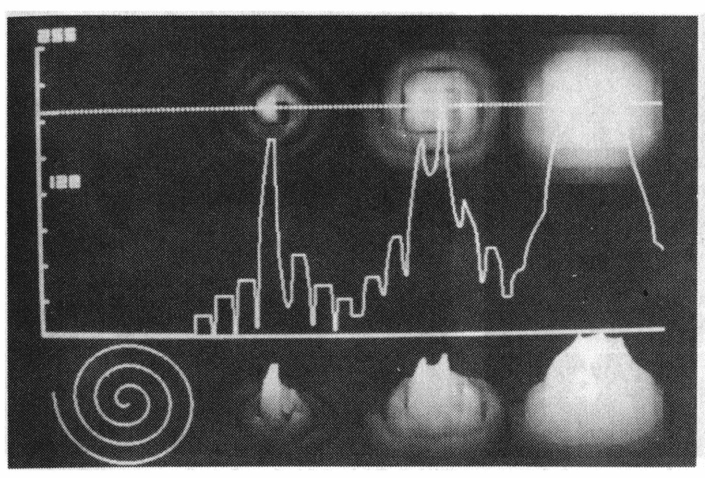


Bild 1. Dichtebild eines spiralförmigen Skeletts
quadratische Fensterform ohne Abstandsbewertung

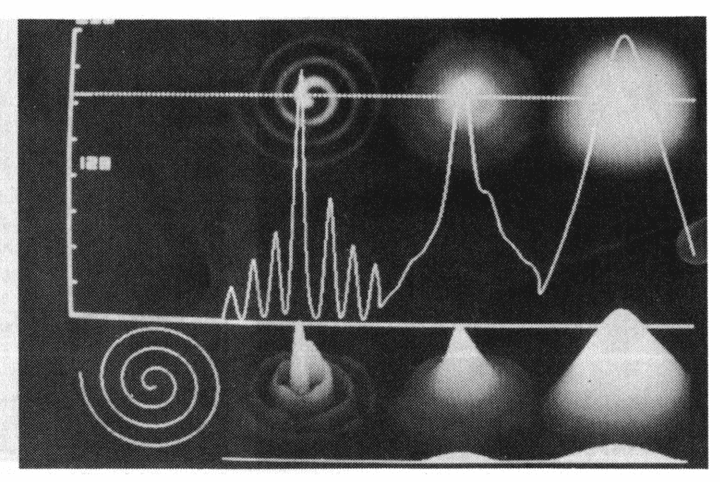


Bild 2. Dichtebild eines spiralförmigen Skeletts
kreissymmetrische Fensterform mit Abstandsbewertung

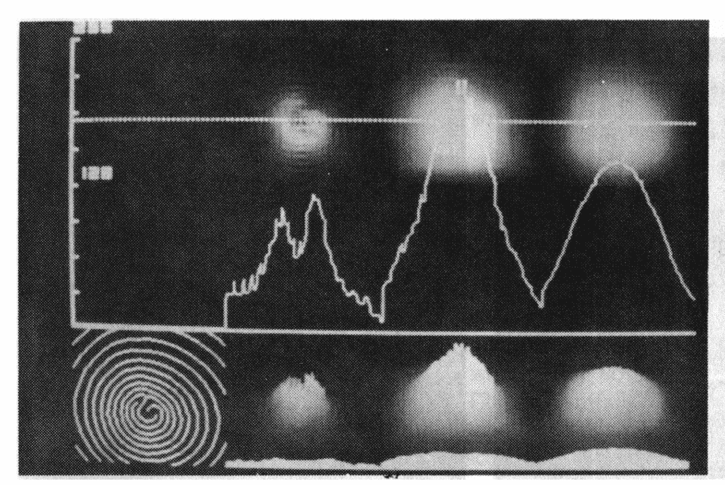


Bild 3. Dichtebild eines Interferogrammskeletts
quadratische Fensterform ohne Abstandsbewertung

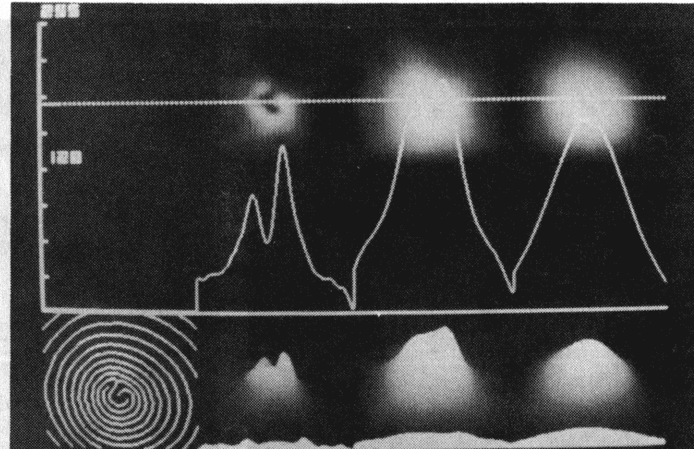


Bild 4. Dichtebild eines Interferogrammskeletts
kreissymmetrische Fensterform mit Abstandsbewertung

ten meist viele, sehr stark gekrümmte Skelettlinien auf. Die Bewertung der Krümmung muß allerdings dem Analyseziel angepaßt werden. So kann man durch geringe Bewertung der Krümmung beispielsweise Gebiete mit lokaler Streifendichte unabhängig von der Streifenform extrahieren oder anderenfalls Konzentrationen stark gekrümmter Streifen finden.
Die Empfindlichkeit der Bewertung gegenüber der digitalen Krümmungsberechnung läßt sich über zwei Schwellwerte K1 und K2 beheben. Unterhalb von K1 werden alle Krümmungen auf den Wert K1 gebracht und K2 beschränkt den Bereich der Krümmungen bei Überschreitung von K2.
Für die Qualität der Dichtebilder ist die Art der Dichteschätzung von großer Bedeutung. So soll die Dichteschätzung eine möglichst kreissymmetrische Fensterform realisieren, um eine lageunabhängige Dichteschätzung zu ermöglichen. Weiterhin sollte bei der Summation der bewerteten kritischen Punkte der Abstand eines Punktes zum Fenstermittelpunkt zusätzlich einbezogen werden, d. h., Punkte nahe dem Fenstermittelpunkt sollen stärker berücksichtigt werden.
Beide Anforderungen lassen sich mit dem Displayprozessor DIP K 6472 durch geeignete Kaskadierung von 2×2- oder 3×3-Fenstern realisieren. Durch die begrenzte Genauigkeit von 8 bit sind gegebenenfalls Normierungen erforderlich.
In einer Untersuchung mit 2×2-Fenstern wurde als günstigste Dichteschätzung ein Algorithmus gefunden, der eine kreissymmetrische Fensterform mit glockenähnlicher Abstandsbewertung verbindet. Gegenüber einer quadratischen Fensterform und konstanter Abstandsbewertung ist etwa die dreifache Rechenzeit erforderlich. Im Gegenzug wird jedoch die Qualität der Dichtebilder wesentlich verbessert, womit eine einfachere und präzisere Auswertung ermöglicht wird.
Das soll anhand der folgenden Bilder nachgewiesen werden. Ausgangspunkt waren drei Skelette, wovon eines synthetisch erzeugt wurde (Spirale). In den Bildern 1 bis 4 sind die Skelette (jeweils unten links), deren Dichtebilder (jeweils oben), die Reliefdarstellungen (jeweils unten) und die Schnitte durch das Dichtebild (Bildmitte) gezeigt. Die Dichte wurde mit drei unterschiedlichen Fenstergrößen (v. l. n. r. zunehmend) ermittelt.
Die Bilder 1 und 2 zeigen die Dichte für die Spirale. In Abhängigkeit von der Fenstergröße tritt das Skelett im Dichtebild mehr oder weniger stark hervor. Zu beachten ist die starke Artefaktbildung beim quadratischen Fenster (Bild 1), die sich in plateauähnlichen Gebilden oder in mehrfachen lokalen Maxima äußert. Dagegen werden beim kreissymmetrischen Fenster mit Abstandsbewertung die Übergänge verschliffen. Das kommt der Auswertung bei der Suche nach lokalen Maxima sehr entgegen.
Beim Dichtebild eines realen Skelettes ist die Zerklüftung des „Gebirges" sehr deutlich sichtbar (Bild 3, kleines Fenster). Sie läßt sich nur durch eine Vergrößerung des Fensters vermindern. Das führt aber meist zur Verdeckung oder Verschmierung von Details (vgl. Bilder 3 und 4). Die gitterartige Strukturierung des Dichtebildes bleibt auch bei größerem Fenster erhalten.
In der Auswertung soll nicht nur durch die Suche nach lokalen Maxima (Überschreiten der kritischen Dichte) auf markante Punkte hingewiesen werden. Es soll aus der Form der „Dichteberge" eine Aussage über die Art der Fehler abgeleitet werden. Die verschiedenen Fehlerarten (siehe oben) zeigen im Dichtebild unterschiedliche Reliefformen. Um diese Formen zu finden, dürfen natürlich keine zusätzlichen Verzerrungen, wie durch das quadratische Fenster, und keine Verschmierungen, wie bei zu großem Fenster, zugelassen werden.
In den Bildern 5 bis 10 sind für ein reales Skelett (jeweils oben links) die Dichtebilder (jeweils unten rechts), deren Reliefdarstellungen (jeweils unten links) und die (senkrechten) Schnitte durch die Dichtebilder (jeweils Mitte links) gezeigt (Bilder 5, 7 und 9: quadratische Fensterform ohne Abstandsbewertung; Bilder 6, 8 und 10:

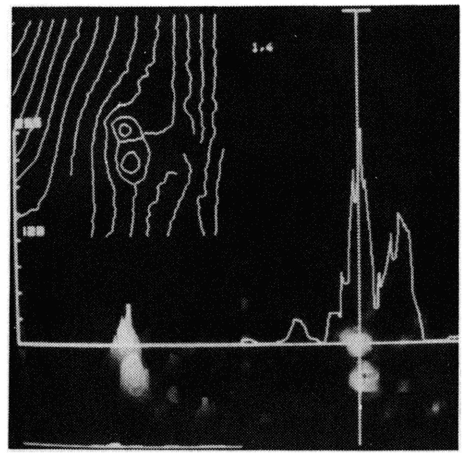


Bild 5. Dichtebild eines Interferogrammskeletts quadratische Fensterform ohne Abstandsbewertung; Fenstergröße: klein

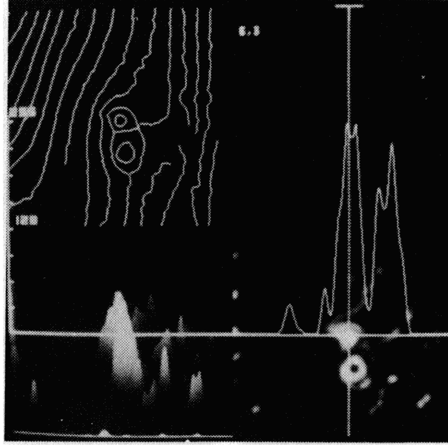


Bild 6. Dichtebild eines Interferogrammskeletts kreissymmetrische Fensterform mit Abstandsbewertung; Fenstergröße: klein

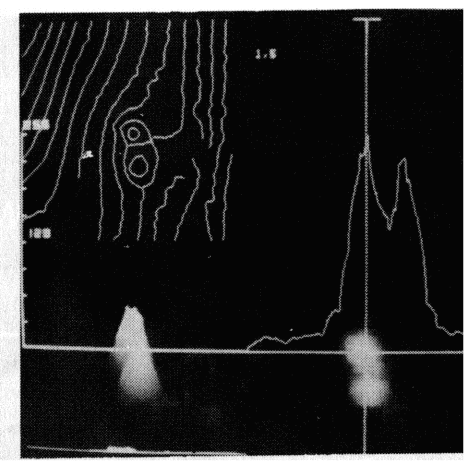


Bild 7. Dichtebild eines Interferogrammskeletts quadratische Fensterform ohne Abstandsbewertung; Fenstergröße: mittel

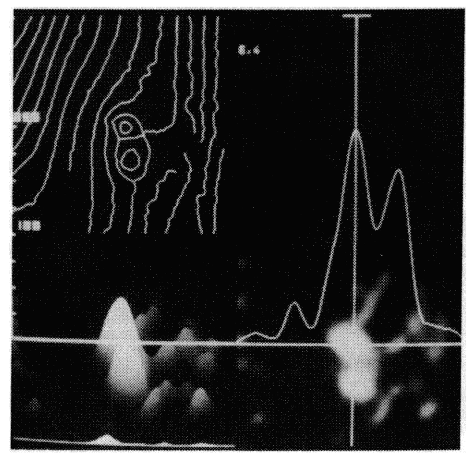


Bild 8. Dichtebild eines Interferogrammskeletts kreissymmetrische Fensterform mit Abstandsbewertung; Fenstergröße: mittel

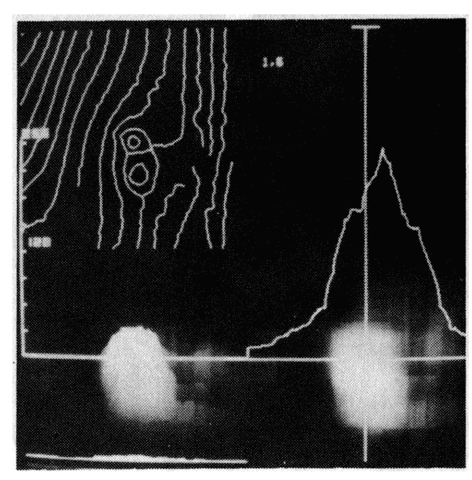


Bild 9. Dichtebild eines Interferogrammskeletts quadratische Fensterform ohne Abstandsbewertung; Fenstergröße: groß

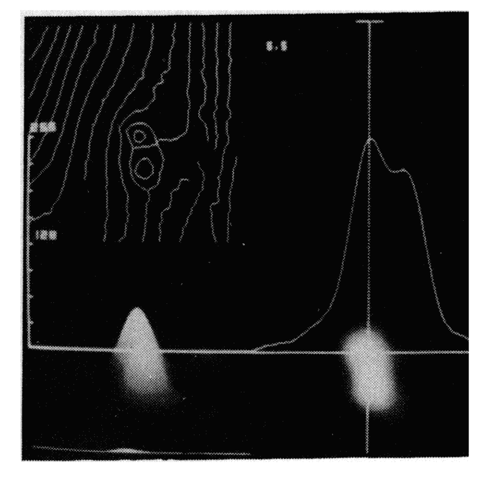


Bild 10. Dichtebild eines Interferogrammskeletts kreissymmetrische Fensterform mit Abstandsbewertung; Fenstergröße: groß

kreissymmetrische Fensterform mit glockenähnlicher Abstandsbewertung; Fenstergröße: Bilder 5, 6 klein, Bilder 7, 8 mittel, Bilder 9, 10 groß).
Man erkennt beim Vergleich deutlich die stärkere Zerklüftung und Strukturierung beim quadratischen Fenster, die unabhängig von der Fenstergröße zur Deformation der Reliefform führt. In der relativ glatten Struktur bei kreissymmetrischen Fenstern ist durch die kontinuierlichen Übergänge die Suche und Beschreibung des Reliefs nach „Bergen", „Tälern", „Sätteln", „Ebenen" usw. wesentlich einfacher und fundierter möglich.

Zusammenfassung

Bildanalysen konzentrieren sich häufig auf markante Bildteile, die besondere Bildeigenschaften beschreiben. Quantifiziert man diese Eigenschaften für die Bildpunkte, so weisen lokale Häufungen von derartigen Punkten auf Abweichungen von der globalen Bildstruktur hin. Im Beitrag wird am Beispiel der Fehlererkennung in Skeletten die Methode der Analyse mit „Dichtebildern" beschrieben und an Beispielen demonstriert. msr 9005

J. Lunze

Wissensbasierte Beratung beim rechnergestützten Entwurf von Automatisierungssystemen

Teil 1: Struktur und Funktionsweise eines Beratungssystems

0. Einleitung

In der Automatisierungstechnik gibt es eine Vielzahl von Entwurfsaufgaben, bei denen der Ingenieur eine Steuereinrichtung finden muß, mit der die für das Zusammenwirken von Prozeß und Steuerung vorgegebenen Funktionsspezifikationen erfüllt werden sollen. Das Entwurfsergebnis kann dabei ein Steuergesetz sein, es kann die Parameter für gerätetechnische Einheiten vorgeben oder es kann in Form einer funktionstüchtigen Software vorliegen.

Für die weitgehend rechnergestützte Lösung derartiger Aufgaben sind in den letzten Jahren vielfältige Entwurfssysteme geschaffen worden. Durch ihren Einsatz wird der Entwurf arbeitsteilig durch einen Rechner und den Ingenieur ausgeführt. Der Rechner übernimmt dabei vor allem komplizierte numerische Operationen und Aufgaben der Datenverwaltung. Der Entwurfsingenieur hat zu entscheiden, welche Algorithmen auf welche Datensätze anzuwenden sind.
Der Dialog zwischen Rechner und Ingenieur vollzieht sich über Indikatoren und Spezifikationen (Bild 1a). Unter Indikatoren werden

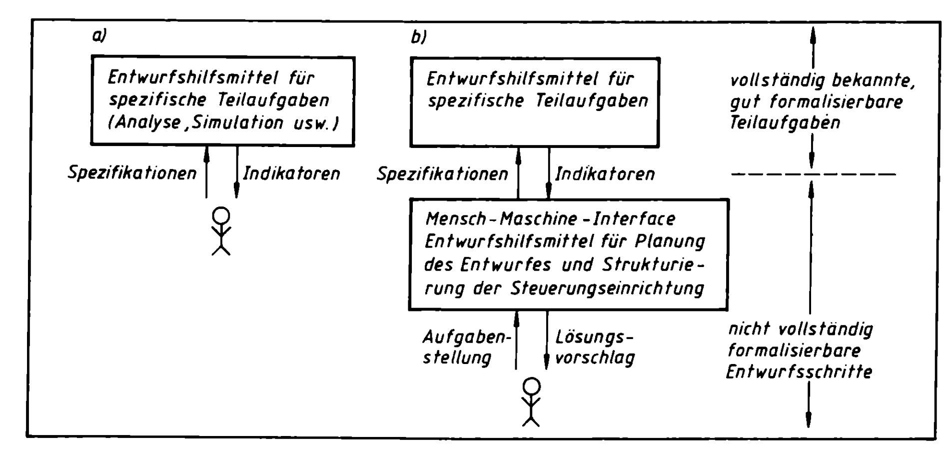


Bild 1.
Struktur von Systemen des rechnergestützten Entwurfes
a) konventionelles Entwurfssystem; b) intelligentes Entwurfssystem

Informationen verstanden, die Eigenschaften des Entwurfsergebnisses beschreiben und dem Entwurfsingenieur einen Eindruck von der Güte des erhaltenen Zwischenergebnisses vermitteln. Beim Reglerentwurf sind dies z. B. der Stabilitätsgrad oder die Übergangsfunktion des geregelten Systems. Der Entwerfer bestimmt den Entwurfsablauf durch Spezifikationen, die den anzuwendenden Algorithmus, dafür notwendige Zusatzparameter sowie die zugehörenden Datensätze bezeichnen.
Typisch für die vom Rechner übernommenen Aufgaben ist, daß der Lösungsweg für diese Aufgaben als Algorithmus vorgegeben ist. Um einen solchen Algorithmus zu erstellen, muß Schritt um Schritt vorausgedacht werden, auf welche Daten der Algorithmus angewendet werden kann, welche Daten bei der Abarbeitung entstehen können und welche Operationen im einzelnen aufeinander folgen.
Es ist offensichtlich, daß diese Form der Rechnerstützung nur für den Teil des Entwurfsprozesses möglich ist, der im voraus im einzelnen bekannt ist. Entwurf als kreative Tätigkeit umfaßt jedoch vielfältige Schritte, die aufgrund von Erfahrungen oder der Intuition des Entwerfers gelöst werden oder spezifisch für eine bestimmte Entwurfsaufgabe sind. Die Ausführung derartiger Schritte rechentechnisch zu unterstützen, erfordert neuartige Methoden für die Darstellung und Verarbeitung des Wissens des Entwurfsingenieurs.
Es ist das Ziel dieses Beitrages, eine Konzeption für ein wissensbasiertes System zur Beratung des Ingenieurs beim rechnergestützten Entwurf von Regelungssystemen zu entwickeln und dessen rechentechnische Realisierung zu beschreiben. Das System soll zwischen dem Entwerfer und dem Entwurfssystem vermitteln und dabei einen Teil der Aufgaben, die heute noch dem Entwerfer zufallen, übernehmen (Bild 1 b).
Entsprechend der in [9] und [13] vertretenen Auffassung, daß es für den *Anwender* wissensbasierter Systeme wesentlich ist, inhaltliche und softwaretechnologische Fragen des Einsatzes von Methoden der künstlichen Intelligenz weitgehend zu trennen, ist der Beitrag in zwei Teile gegliedert. Im ersten Teil werden die Überführung der organisatorischen Probleme des Entwurfes in ein Wissensverarbeitungsproblem und dessen Lösung mit Hilfe eines regelbasierten Systems erläutert. Der zweite Teil ist der Realisierung des Systems als logisches Programm vorbehalten. Beide Teile verfolgen gleichermaßen das Ziel, die Lösung des fachwissenschaftlichen Problems mit Methoden der Wissensverarbeitung im Detail zu erläutern. Für den Kern des Beratungssystems wird in bezug auf Konzeption und Implementierung eine vollständige Lösung angegeben, die den Leser in die Lage versetzt, sich mit den grundlegenden Funktionen des Beratungssystems und seiner Realisierung unter Verwendung einer repräsentativen Sprache der künstlichen Intelligenz vertraut zu machen.

Prof. Dr. sc. techn. JAN LUNZE (37) studierte von 1970 bis 1974 an der Sektion Technische und Biomedizinische Kybernetik der TH Ilmenau. Seit 1974 wissenschaftlicher Mitarbeiter am Zentralinstitut für Kybernetik und Informationsprozesse der AdW der DDR, Bereich Regelungssysteme, Dresden. 1980 Promotion A (Entwurf dezentraler Regler), 1983 Promotion B (Robuste Mehrgrößenregelung). Seit 1977 nebenamtliche Lehrtätigkeit. 1985 Verleihung der Facultas docendi für Automatisierungstechnik durch die TU Dresden. 1988 Ernennung zum Akademie-Professor. Vorsitzender des FA „Theorie der automatischen Steuerung" der WGMA.

Unter einem „wissensbasierten System" soll im folgenden ein System verstanden werden, bei dem die Wissensverarbeitung eine dominierende Rolle spielt. Im Gegensatz dazu wird in der Literatur mit „Expertensystem" meist ein abgeschlossenes (*stand-alone*) System bezeichnet. In der Produktionsautomatisierung sind wissensbasierte Systeme meist nur Bestandteile einer größeren Automatisierungseinrichtung. Für sie sind deshalb — wie auch bei dem hier beschriebenen Beratungssystem — nicht alle Komponenten eines Expertensystems vollständig ausgebaut.
Erste Ansätze für eine wissensbasierte Unterstützung des Reglerentwurfes wurden in [11] [12] [14] und [16] für lineare Regler, in [7] für robuste Regler sowie in [15] für nichtlineare Regler beschrieben. Diese Arbeiten weisen auf die Notwendigkeit des Einsatzes von wissensbasierten Systemen zur Organisation des Entwurfsablaufes hin, geben jedoch keine detaillierten Hinweise auf die von den Autoren beabsichtigte Art der Realisierung. In [12] wird ausführlich auf die Rückwirkungen eingegangen, die eine wissensbasierte Erweiterung der Entwurfssoftware auf die Gestaltung der Entwurfsalgorithmen hat. Es werden Modifikationen bekannter Indikatoren für die Stabilität, das Eingangs-Ausgangs-Verhalten und die Robustheit von Mehrgrößenregelungen angegeben und drei Entwurfsverfahren entwickelt, deren Einsatzgebiete gut bekannt sind und die in zweckmäßiger Weise in der wissensbasierten Komponente festgehalten und ausgenutzt werden können.

1. Entwurfsaufgabe

In dieser Arbeit soll der rechnergestützte Entwurf linearer Regler als Beispiel dienen, um ein wissensbasiertes Beratungssystem für Entwurfsaufgaben zu entwickeln. Der im folgenden vorangestellte Abriß zu den theoretischen Grundlagen und zum gegenwärtigen Stand der Entwurfssoftware charakterisiert das allgemeine Entwurfsvorgehen und bildet die Basis dafür, daß die anschließend entwickelte Konzeption für ein Beratungssystem auch auf andere Entwurfsprobleme übertragen werden kann.
Beim Parameterentwurf des Reglers wird vorausgesetzt, daß die Struktur der Regelung bekannt ist und ein mathematisches Modell des Prozesses vorliegt. Hier soll der Entwurf im Zeitbereich betrachtet werden. Für die Regelstrecke sei ein Zustandsraummodell

$$\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u} + \boldsymbol{E}\boldsymbol{z}$$
$$\boldsymbol{y} = \boldsymbol{C}\boldsymbol{x} + \boldsymbol{D}\boldsymbol{u} + \boldsymbol{F}\boldsymbol{z} \qquad (1)$$

verfügbar, bei dem $\boldsymbol{x}$, $\boldsymbol{u}$, $\boldsymbol{y}$ und $\boldsymbol{z}$ die n-, m-, r- bzw. p-dimensionalen Vektoren der Systemzustandsgrößen, Stellgrößen, Regelgrößen bzw. Störgrößen darstellen. Ein dynamischer Regler hat die Form

$$\dot{\boldsymbol{x}}_r = \boldsymbol{A}_r\boldsymbol{x}_r + \boldsymbol{B}_y\boldsymbol{y} + \boldsymbol{B}_v\boldsymbol{v}$$
$$\boldsymbol{u} = \boldsymbol{K}_r\boldsymbol{x}_r + \boldsymbol{K}_y\boldsymbol{y} + \boldsymbol{K}_v\boldsymbol{v}\,, \qquad (2)$$

wobei $\boldsymbol{v}$ der r-dimensionale Vektor der Führungsgrößen ist. Das Entwurfsziel besteht in der Festlegung der in die Gleichung (2) eingehenden Parameter in einer solchen Weise, daß das geregelte

System (1), (2) vorgegebene Güteforderungen erfüllt, die typischerweise in drei Komplexen zusammengefaßt werden können:

(I) Das geregelte System (1), (2) soll stabil sein.

(II) Für vorgegebene Klassen von Stör- und Führungssignalen soll der Regelfehler asymptotisch verschwinden, d. h. $y(t) - v(t) \to 0$ für $t \to \infty$.

(III) Das dynamische Verhalten soll vorgegebene Kennwerte haben (z. B. soll das Überschwingen der Führungsübergangsfunktion eine gegebene obere Schranke nicht überschreiten).

2. Grundlagen der Entwurfsverfahren

Die folgenden Ergebnisse aus der Theorie der automatischen Steuerung bilden die Grundlage für die im Abschn. 3. zusammengestellten Moduln eines Programmsystems für die rechnergestützte Lösung der angegebenen Entwurfsaufgabe. Die Entwurfsforderung (II) wird bei geeigneter Wahl der Reglerstruktur erfüllt, wobei im folgenden nur P-Regler (für impulsförmige Stör- und Führungssignale) und PI-Regler (für sprungförmige Signale) betrachtet werden. PI-Regler können eingesetzt werden, wenn die Statikmatrix

$$K_s = D - CA^{-1}B \tag{3}$$

der Regelstrecke (1) vollen Rang hat [8]. Das geregelte System (1), (2) kann in die Form

$$\begin{aligned} \dot{\bar{x}} &= \bar{A}\bar{x} + \bar{B}r + \bar{E}z \\ y &= \bar{C}\bar{x} + \bar{D}v + \bar{F}z \end{aligned} \tag{4}$$

überführt werden. Es ist stabil, wenn sämtliche Eigenwerte der Matrix $\bar{A}$ negativen Realteil haben. Die Führungsübergangsfunktion

$$H_v(t) = \bar{K}_s - \bar{C}\bar{A}^{-1} \exp(\bar{A}t)\,\bar{B} \tag{5}$$

dient zur Beurteilung der Dynamik des geschlossenen Kreises (4) entsprechend den Güteforderungen (III).

Für eine proportionale Ausgangsrückführung hat (2) die Form

$$u = K_y y \tag{6}$$

Zustandsrückführungen werden nur dann als realisierbar betrachtet, wenn ein Zustandsbeobachter

$$\begin{aligned} \dot{x}_b &= A_b x_b + B_b u + E_b y \\ \hat{x} &= C_b x_b + D_b u + F_b y \end{aligned} \tag{7}$$

vorhanden ist und anstelle des Zustandes x der Näherungswert $\hat{x}$ zurückgeführt werden kann

$$u = K\hat{x}\,. \tag{8}$$

Der Regler (2) besteht dann aus den zwei Bestandteilen (7) und (8). PI-Regler haben die Beschreibung

$$\begin{aligned} \dot{x}_r &= y - v \\ u &= K_I x_r + K_P(y - v)\,. \end{aligned} \tag{9}$$

Sie werden dadurch entworfen, daß die Regelstrecke (1) zunächst durch den dynamischen Anteil des Reglers (erste Zeile in (9)) erweitert wird, wodurch die I-erweiterte Regelstrecke

$$\begin{aligned} \begin{pmatrix} \dot{x} \\ \dot{x}_r \end{pmatrix} &= \begin{pmatrix} A & 0 \\ C & 0 \end{pmatrix} \begin{pmatrix} x \\ x_r \end{pmatrix} + \begin{pmatrix} B \\ D \end{pmatrix} u + \begin{pmatrix} E \\ F \end{pmatrix} z + \begin{pmatrix} 0 \\ -I \end{pmatrix} v \\ \begin{pmatrix} y - v \\ x_r \end{pmatrix} &= \begin{pmatrix} C & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} x \\ x_r \end{pmatrix} + \begin{pmatrix} D \\ 0 \end{pmatrix} u + \begin{pmatrix} F \\ 0 \end{pmatrix} z + \begin{pmatrix} -I \\ 0 \end{pmatrix} v \end{aligned} \tag{10}$$

entsteht. Die in der zweiten Zeile von (9) stehende Rückführung ist dann eine Ausgangsrückführung

$$u = (K_P\ K_I) \begin{pmatrix} y - v \\ x_r \end{pmatrix}. \tag{11}$$

Für den Entwurf der Regler (6), (8) und (11) können analytische oder heuristische Verfahren eingesetzt werden. Da die oft verbal gegebenen Entwurfsforderungen (III) nicht bzw. nur schwer analytisch verarbeitbar sind, beruhen die meisten Entwurfsverfahren auf Ersatzkriterien. So müssen bei Verfahren der Polverschiebung Werte für die Pole des geschlossenen Kreises und beim Optimalreglerentwurf Wichtungsmatrizen des Gütefunktionals

$$\int x'Qx + u'Ru \,\mathrm{d}t \to \min_K \tag{12}$$

vorgegeben werden. Heuristische Einstellungen für PI-Regler können z. B. mit Hilfe der Statikmatrix (4) entsprechend

$$K_I = -aK_s^{-1}\,, \qquad K_P = -bK_s^{-1} \tag{13}$$

vorgenommen werden, wobei die Einstellparameter a und b aus Kenntnissen über das Übergangsverhalten der Strecke (1) bestimmt werden [8].

Diese Erläuterungen beschränken sich auf den Teil von Entwurfsverfahren, die als Beispiel im weiteren betrachtet werden. Eine vollständige Übersicht kann z. B. [6] entnommen werden.

3. Rechnergestützter Entwurf linearer Regler

Programmsysteme für den rechnergestützten Entwurf von Regelungssystemen sind i. allg. so aufgebaut, daß für jeden der beim Entwurf auszuführenden Teilschritte ein Modul zur Verfügung gestellt wird (siehe z. B. [5]). Ein aus 12 Moduln bestehendes Entwurfssystem ist in Tafel 1 angegeben.

In der rechentechnischen Implementierung stellen die Moduln abarbeitbare Programme dar, die der Nutzer über das Betriebssystem aufruft. Die Vermittlung der Modellparameter erfolgt über die Angabe der Namen der Dateien, aus denen die Parameter gelesen werden können bzw. in die das Ergebnis geschrieben werden soll.

Diese, vom Reglerentwurf her bekannte, aber auch auf andere Entwurfsaufgaben zutreffende Form der Rechnerstützung ist die Motivation für die Entwicklung eines Beratungssystems, das auf folgende **Charakteristika des rechnergestützten Entwurfes** Rücksicht nimmt:

* Das *Entwurfssystem* besteht aus einer Vielzahl von Moduln, die unter bestimmten Anwendungsbedingungen aufgerufen werden können und durch die dann für gegebene Systemparameter ein Entwurfsschritt ausgeführt wird.
* Die Organisation des Entwurfsprozesses erfolgt durch den *Entwurfsingenieur* in zwei Schritten:
  (1) Entwurfsplanung: Es wird eine Folge von Entwurfsschritten zusammengestellt, mit der die gegebene Entwurfsaufgabe gelöst werden kann.
  (2) Entwurfsdurchführung: Durch Aufruf der Moduln und Vermittlung der Parametersätze wird die Arbeitsweise des Entwurfssystems gesteuert.

Ein **Entwurfsplan** ist eine Folge von Modulnamen, bei der gesichert ist, daß bei der Abarbeitung des Planes zum Zeitpunkt des Modulaufrufes alle in den Anwendungsbedingungen des Moduls angegebenen Datensätze verfügbar sind. Der Entwurfsplan wird vom Entwurfsingenieur aufgrund dessen Kenntnisse über die im Entwurfssystem zur Verfügung stehenden Moduln zusammengestellt. Dabei wird lediglich auf die *Existenz* bestimmter Datensätze in einem betrachteten Entwurfsschritt, nicht jedoch auf die in den Datensätzen vorhandenen numerischen Werte, Rücksicht genommen. Dieser Tatsache soll im weiteren dadurch Rechnung getragen werden, daß von einer „strukturellen Betrachtungsweise" gesprochen wird.

Für einen gegebenen Entwurfsplan kann die Entwurfsdurchführung im Prinzip automatisch erfolgen. Der Entwurfsingenieur wird den Entwurfsplan aber nur soweit realisieren, wie die erhaltenen Zwischenergebnisse den erwarteten entsprechen. Aus zwei Gründen ist der Entwurf kein im voraus vollständig planbarer Prozeß:

* Die Entwurfsaufgabe ist nicht exakt formuliert, sondern durch Forderungen gekennzeichnet, die sich widersprechen bzw. nur Richtwerte für bestimmte Kennwerte enthalten. Gesucht wird eine akzeptable Kompromißlösung, die alle Entwurfsforderungen angemessen berücksichtigt.
* Die Wirkung der Moduln auf das Entwurfsobjekt ist nicht exakt bekannt. Da der Entwurfsschritt mit einem Ersatzkriterium ausgeführt, das Entwurfsergebnis aber anhand der ursprünglich gege-

Tafel 1. Moduln des Entwurfssystems

Anwendungsvoraussetzungen	Ergebnis	Entwurfsschritt
Modell (1) oder (4)	Stabilitätsaussage	Stabilitätstest
Modell (1) der Strecke	Statikmatrix K_s aus (3)	Statikberechnung
Statikmatrix K_s aus (3)	Existenzaussage für PI-Regler	Existenzprüfung PI-Regler
Modell (4) des Kreises, Stabilitätsaussage für Kreis	Übergangsfunktion $H_v(t)$ (5)	Simulation
Modell (1) der Strecke und Modell (2) des Reglers	Modell (3) des Regelkreises	Berechnung des Modells des Kreises
Übergangsfunktion $H_v(t)$ (5)	Bild der Übergangsfunktion	Grafik
Modell (1) der Strecke	I-erweiterte Strecke (10)	I-Erweiterung der Strecke
Modell (1) der Strecke	Beobachter (7)	Beobachterentwurf
Modell (1) der Strecke und Beobachter (7)	Zustandsregler (7), (8)	Polzuweisung durch Zustandsrückführung
Modell (1) der Strecke und Beobachter (7)	Zustandsregler (7), (8)	Optimalreglerentwurf
Modell (1) der Strecke und Stabilitätsaussage für Strecke (1)	Ausgangsrückführung (6)	Ausgangsrückführung
Modell (1) der Strecke und Statikmatrix K_s aus (4) und Existenzaussage für PI-Regler	PI-Reglerparameter (13)	heuristische Einstellung des PI-Reglers

benen Spezifikationen für den geschlossenen Kreis beurteilt wird, kann nicht exakt vorhergesagt werden, welcher Entwurfsmodul mit welchen Parametern das gegebene Problem zu lösen hilft.

4. Zielstellung des Beratungssystems

Das Beratungssystem soll den Ingenieur bei der *Organisation* des Entwurfsprozesses unterstützen. Das betrifft im einzelnen:

- Die Entwurfsplanung, bei der das Beratungssystem alternative Pläne ermitteln und dem Ingenieur zur Auswahl anbieten soll
- die Entwurfsdurchführung, bei der das Beratungssystem die Moduln des Entwurfssystems aufrufen und mit den notwendigen Daten versorgen soll
- die Buchführung, bei der das Beratungssystem die Entwurfsschritte so registrieren soll, daß es jederzeit Auskunft über den noch nicht realisierten Entwurfsplan sowie die bereits durchlaufenen Entwurfsschritte geben kann.

Um diese Zielstellung erfüllen zu können, muß das Beratungssystem das Wissen „besitzen" und verarbeiten, mit Hilfe dessen bisher der Nutzer des Entwurfssystems die organisatorischen Aufgaben gelöst hat. Dieses Wissen umfaßt

(W1) Wissen über die Anwendungsbedingungen und Wirkungen der Entwurfsmoduln: Dieses Wissen ist in Tafel 1 zusammengestellt.

(W2) Wissen über die Sinnfälligkeit von Entwurfsschrittfolgen in bezug auf eine gegebene Entwurfsaufgabe: Dieses Wissen bildet die Grundlage für den in den Abschn. 6. und 12. beschriebenen Planungsalgorithmus.

(W3) Wissen über den aktuellen Bearbeitungsstand der Entwurfsaufgabe: Dieses Wissen ist im Zustand des Entwurfsprozesses zusammengefaßt (Abschn. 5.) und im Arbeitsspeicher des Beratungssystems zusammengestellt (Abschn. 11.).

(W4) Wissen über die Verarbeitung des in (W1) bis (W3) angegebenen Wissens: Dieses Wissen ist die Basis für den Aufbau des Beratungssystems. Es bestimmt Struktur und Zusammenwirken der einzelnen Komponenten des wissensbasierten Systems.

Das Beratungssystem übernimmt Funktionen, die nicht die Ausführung der Entwurfsschritte selbst, sondern die Organisation des Entwurfsprozesses betreffen. Das wird im Bild 2 veranschaulicht. Während die Moduln des Entwurfssystems auf der funktionalen Ebene arbeiten und Teilschritte des Entwurfes ausführen, liegt die Wirkung des Beratungssystems auf der strukturellen Ebene. Im Bild 2 kennzeichnen Knoten den Bearbeitungsstand, wie er durch das Vorhandensein bestimmter Datensätze zu einem bestimmten Zeitpunkt beschrieben ist. Kanten veranschaulichen die Umformung von Zwischenergebnissen durch Algorithmen des Entwurfssystems. Für eine gegebene Entwurfsaufgabe werden Wege zwischen den gegebenen Spezifikationen (Startknoten) und einer diesen Forderungen genügenden Regelung (Zielknoten) gesucht und dem Entwurfsingenieur vorgeschlagen.

Im weiteren wird untersucht, wie diese Idee formalisiert und durch ein regelbasiertes System realisiert werden kann.

5. Zustandsraumdarstellung von Wissensverarbeitungsproblemen

Als Grundlage für die Interpretation der Aufgaben des Beratungssystems als Problem der Wissensverarbeitung muß eine Zustandsraumdarstellung des Wissensverarbeitungsproblems gefunden wer-

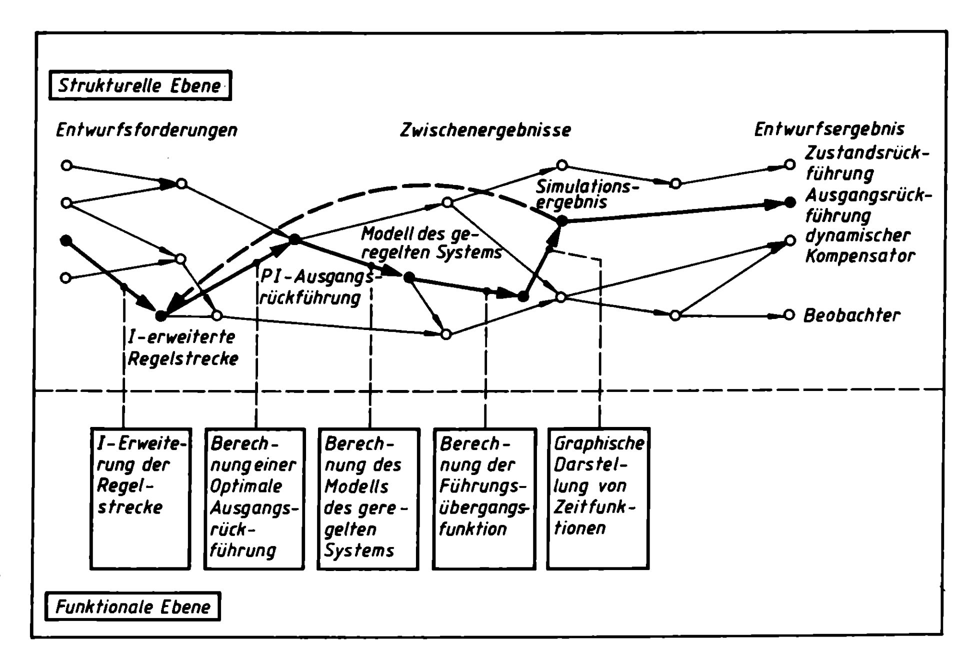


Bild 2.
Zusammenhang zwischen Entwurfsplanung auf der strukturellen Ebene und Entwurfsdurchführung auf der funktionalen Ebene

den. Diese Darstellung wurde für Aufgaben entwickelt, die mit Hilfe von Regeln der Form

$$\text{IF } A_1, A_2, \dots, A_n \text{ THEN } B \qquad (14)$$

gelöst werden sollen [1]. Die Regeln enthalten n Bedingungen A_i und eine Schlußfolgerung B. Sie können nur dann angewendet werden, wenn alle n Bedingungen A_i gleichzeitig erfüllt sind. Ist dies der Fall, so wird die Schlußfolgerung B gezogen.

Ein **Wissensverarbeitungsproblem** ist durch drei Elemente beschrieben:

* **Fakten,** die die als gegeben vorausgesetzten Tatsachen beschreiben
* **Regeln** der Form (14), die Zusammenhänge zwischen Fakten beschreiben
* **Anfragen,** die das Ziel der Wissensverarbeitung formulieren.

Gesucht sind die Antworten auf die gestellte Anfrage, die unter Zuhilfenahme der Regeln aus den bekannten Fakten abgeleitet werden können.

Als **Zustand** wird das im betrachteten Bearbeitungsschritt zu lösende Problem bezeichnet. Während der Anfangszustand durch die Fakten und die Beschreibung des Zieles gegeben ist, gehen in den Zustand i. allg. auch die während des Lösungsprozesses erhaltenen Informationen über erfolglose bzw. noch nicht getestete Lösungswege ein. Mit anderen Worten: Der Zustand verkörpert all jene Informationen, die es ermöglichen, nach einem Abbruch des Lösungsprozesses die Suche fortzusetzen. Insofern ist die Begriffsbestimmung des Zustandes ähnlich der von der Theorie dynamischer Systeme her bekannten Definition. Dort vereinigt der Zustand ja auch alle für die Bestimmung des zukünftigen Verhaltens des Systems bei vorgegebener Eingangsgröße notwendigen Informationen.

Der Zustand eines Entwurfsproblems wird durch das Vorhandensein von Datensätzen mit einem bestimmten Inhalt charakterisiert. So ist z. B. der Zustand nach Anwendung der Moduln für den Beobachterentwurf und die Polzuweisung auf das gegebene Modell der Regelstrecke durch die Existenz der Datensätze für das Streckenmodell, den Beobachter und die Zustandsrückführung gekennzeichnet. Für die Zustandsraumdarstellung bilden die Fakten die **Anfangsbedingungen** für die Lösung des Problems. Beim Entwurf sind sie durch die Kenntnisse des Entwerfers bestimmt, z. B.

Anfangsbedingung S_1:
Modell (1) der Strecke vorhanden (15)

Wurden bereits zu einem früheren Zeitpunkt Entwurfsschritte vorgenommen, so kann auch folgende Anfangsbedingung gegeben sein:

Anfangsbedingung S_2:
Stabilitätsaussage über die Strecke vorhanden
I-erweiterte Strecke (10) vorhanden (16)

Der Zielzustand wird durch **Prädikate** definiert, die anzeigen, ob ein Zustand die Antwort auf die Anfrage gibt und das Problem somit gelöst ist. Bei einer strukturellen Betrachtung besteht das Ziel des Entwurfes in der Erzeugung derjenigen Datensätze, mit denen der Entwerfer den Erfolg bzw. Mißerfolg der Entwurfsdurchführung erkennen und gegebenenfalls neue Entwurfsaufgaben stellen kann. Dementsprechend werden die Entwurfsforderungen (I) und (III) bei Existenz einer Stabilitätsaussage für den geschlossenen Kreis bzw. bei Vorhandensein einer grafischen Darstellung der Führungsübergangsfunktion als erfüllt betrachtet. Die Entwurfsforderung (II) geht in die Forderung nach Verwendung einer PI- bzw. proportionalen Rückführung ein. Als Ziele können z. B. vorgegeben werden

Zielzustand Z_1:
P-Regler vorhanden
Stabilitätsaussage für den P-Regelkreis vorhanden (17)

Zielzustand Z_2:
PI-Regler vorhanden
Stabilitätsaussage für den PI-Regelkreis vorhanden
Bild der Übergangsfunktion vorhanden (18)

Die Regeln (14) beschreiben **Operatoren** mit den im IF-Teil stehenden Anwendungsbedingungen $A_1, \dots, A_n$ und der im THEN-Teil stehenden Aktion B. Durch Anwendung der Operatoren wird der Problemzustand in einen neuen Zustand transformiert. Für das Entwurfssystem können die Angaben in Tafel 1 folgendermaßen als Regeln interpretiert werden:

Stabilitätstest:
IF Modell (3) des Kreises vorhanden
THEN Ermittle Stabilitätsaussage über den Kreis (19)

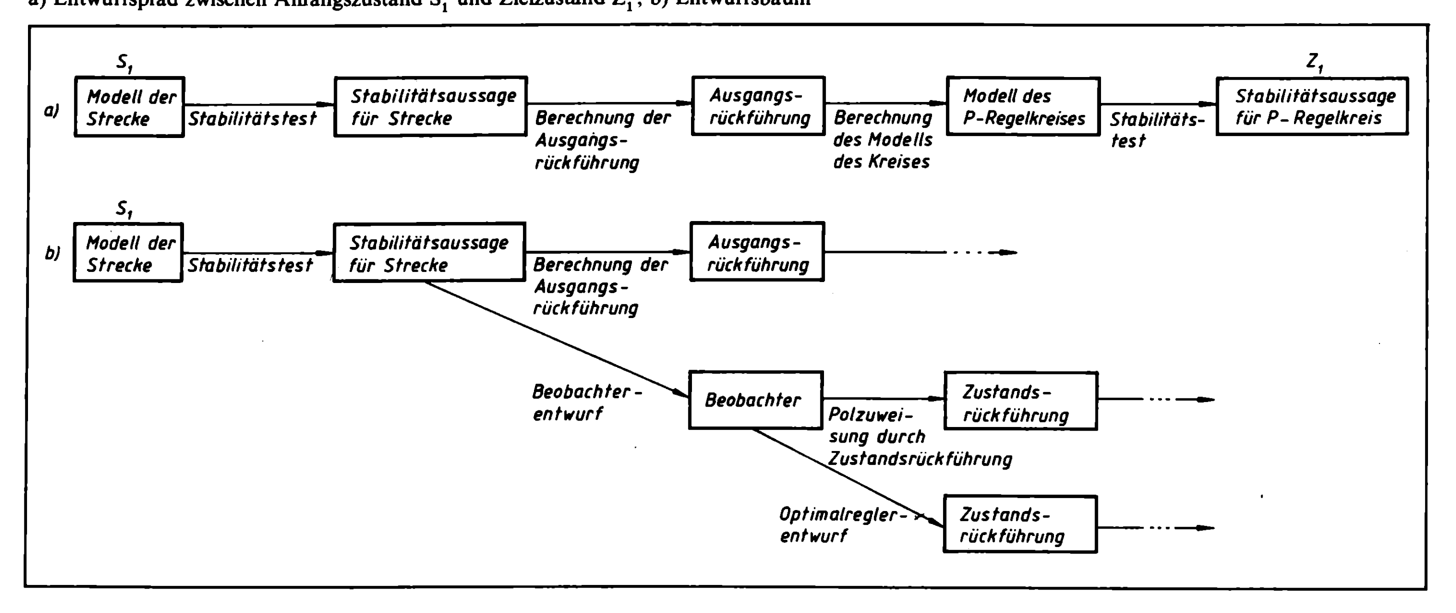
I-Erweiterung der Strecke:
IF Modell (1) der Strecke vorhanden
THEN Ermittle I-erweiterte Strecke (10) (20)

Berechnung des Modells des Kreises:
IF Modell (1) der Strecke vorhanden
AND Modell (2) des Reglers vorhanden
THEN Ermittle Modell (3) des Kreises (21)

Jeder Modul des Entwurfssystems stellt also im Sinne der Wissensverarbeitung einen Operator dar, dessen Wirkung B in der Schaffung eines neuen Datensatzes mit bestimmtem Inhalt, beispielsweise des Datensatzes „I-erweiterte Strecke", besteht.

Außer den die Anwendbarkeit der Entwurfsmoduln beschreibenden Regeln müssen noch die beim Entwurf verwendeten Begriffe in ihrer Bedeutung zueinander notiert werden. In der folgenden Taxonomie stellt der jeweils höher angeordnete Begriff die Bezeichnung für eine Klasse dar, zu der die darunter stehenden Begriffe gehören.

Bild 3. Grafische Veranschaulichung des Entwurfsprozesses
a) Entwurfspfad zwischen Anfangszustand S_1 und Zielzustand Z_1; b) Entwurfsbaum



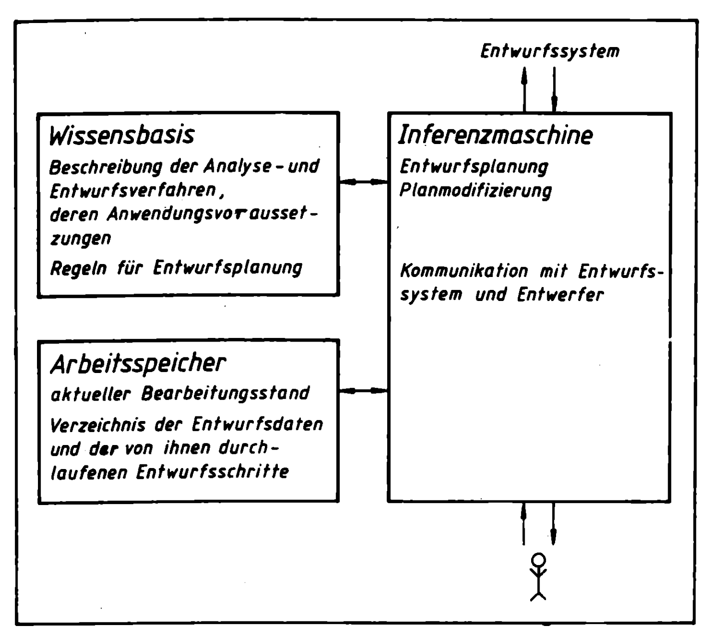
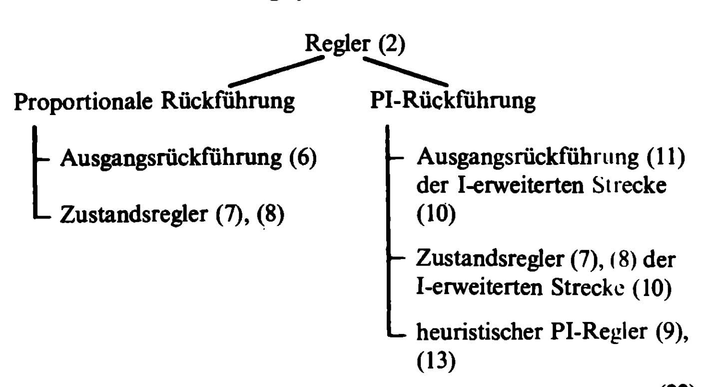


Bild 4. Struktur des Beratungssystems



(22)

Mit der Zustandsraumdarstellung kann der Entwurfsprozeß in der im Bild 2 bereits angedeuteten Weise grafisch veranschaulicht werden. Der Zustand entspricht einem Knoten des Graphen, die Anwendung eines Moduls einer Kante (Bild 3). In jedem Knoten können die den Zustand charakterisierenden Datensätze eingetragen werden. Der Einfachheit halber wurden im Bild 3 jedoch nur die jeweils neuen Daten vermerkt.

Die Abarbeitung eines Entwurfsplanes ergibt einen Entwurfspfad (Bild 3a). Aufgrund der im Abschn. 3. beschriebenen Merkmale des Entwurfsprozesses wird mit der Ausführung eines solchen Entwurfspfades das Entwurfsproblem noch nicht vollständig gelöst sein. Beispielsweise kann ungenügendes dynamisches Verhalten des geschlossenen Kreises den Entwurfsingenieur veranlassen, zu versuchen, anstelle einer Ausgangsrückführung einen Beobachter und eine Zustandsrückführung einzusetzen. Das bedeutet, daß er zu einem früher durchlaufenen Entwurfszustand zurückgeht (in diesem Falle zum Anfangszustand S_1), einen neuen Entwurfsplan aufstellt und auf einem anderen Weg zum Entwurfsziel „P-Regler" gelangt. Der Entwurfsvorgang schlägt sich dann in einer Baumstruktur des Graphen nieder (Bild 3b).

6. Entwurfsplanung als Problem der Wissensverarbeitung

Die Entwurfsplanung stellt die Aufgabe, eine Folge von Operationen zu finden, durch die der Anfangszustand in den Zielzustand überführt wird. Diese Aufgabe soll durch Rückwärtsverkettung (*backward chaining*) der die Operatoren beschreibenden Regeln der Form (19) bis (21) gelöst werden.

Bei der Rückwärtsverkettung wird angenommen, der Zielzustand, z. B. Z_1, sei erreicht. Die Anwendung des Operators „Stabilitätstest" (19), dessen Wirkung in der Erzeugung des Datensatzes „Stabilitätsaussage für den P-Regelkreis" besteht, führt die Entwurfsaufgabe auf das Problem zurück, den Anfangszustand S_1 in einen Zustand zu überführen, in dem das Modell des P-Regelkreises vorhanden ist (Bild 3a). Im Sprachgebrauch der Wissensverarbeitung heißt das, daß das Ziel Z_1 durch das Ziel „Modell des P-Regelkreises vorhanden" ersetzt wird. Durch die Regel (21) kann dieses Ziel in die Teilziele „Modell der Strecke vorhanden" und „Modell des P-Reglers vorhanden" überführt werden. Da das Modell der Strecke gegeben ist („ein Fakt ist"), verbleibt als einziges Ziel, einen P-Regler zu entwerfen. Das Problem der Entwurfsplanung ist gelöst, wenn der erzeugte Zustand keine offenen Teilziele mehr enthält. Bei der Entwurfsdurchführung sind die Moduln in der Reihenfolge anzuwenden, in der die Regeln vom Ausgangszustand zum Zielzustand aufeinanderfolgen, also in umgekehrter Reihenfolge ihrer Anwendung bei der Planung.

7. Struktur eines regelbasierten Beratungssystems

Für das Beratungssystem wird die im Bild 4 gezeigte Struktur verwendet, die für regelbasierte Systeme typisch ist. Das für die Entwurfsorganisation notwendige Wissen (Entwurfsziel, Entwurfsregeln, Entwurfszustand) und die Regeln für die Verarbeitung dieses Wissens sind in getrennten Bestandteilen eines regelbasierten Systems enthalten [9] und [13].

Die **Wissensbasis** umfaßt die den rechnergestützten Entwurf auf der strukturellen Ebene beschreibenden Fakten und Regeln aus Tafel 1. Sie ist unabhängig vom betrachteten Entwurfsproblem und wird auch nicht bei der Arbeit mit dem Beratungssystem verändert.

Im **Arbeitsspeicher** sind die ein konkretes Entwurfsproblem beschreibenden Fakten abgelegt. Es sind dies

- zur Beschreibung der Entwurfsaufgabe:
  die Entwurfsziele (z. B. Zielzustand Z_1 aus (17))
  die gegebenen Daten (z. B. Anfangszustand S_1 aus (15))
- zur Beschreibung des Lösungsweges:
  der Teil des Planes, der noch nicht abgearbeitet ist
- zur Beschreibung des aktuellen Bearbeitungsstandes:
  die durchgeführten Entwurfsschritte einschließlich der dabei erzeugten Datensätze (z. B. Datei-Namen).

Diese Informationen verändern sich mit jedem ausgeführten Planungs- und Entwurfsschritt.

Die **Inferenzmaschine** verarbeitet bei der Entwurfsplanung und -durchführung das in beiden Speichern enthaltene Wissen und aktualisiert den Arbeitsspeicher. Außerdem muß die Interferenzmaschine den Dialog mit dem Nutzer auf der einen und dem Entwurfssystem auf der anderen Seite führen (dies könnte auch in einer getrennten Dialogkomponente zusammengefaßt werden).

Die Realisierung eines derartigen regelbasierten Beratungssystems wird im zweiten Teil des Beitrages erläutert.

8. Inhaltliche Charakteristika des Beratungssystems

Eine Entwurfsaufgabe zu lösen heißt, im Raum aller möglichen Verkettungen der Moduln des Entwurfssystems einen solchen Entwurfsweg zu finden, auf dem aus den Ausgangsdaten die gesuchte Lösung erzeugt wird. Bei der Entwurfsplanung ermittelt das Beratungssystem einen bzw. alle derartigen Wege; bei der Entwurfsdurchführung unterstützt es die Datenverwaltung und Protokollierung.

Im Unterschied zwischen dem Raum aller möglichen Verknüpfungen der Entwurfsmoduln und dem Entwurfsweg spiegelt sich der Unterschied zwischen der universellen Einsetzbarkeit der Moduln und deren zweckentsprechender Anwendung für eine gegebene Entwurfsaufgabe wider. Bisher mußte der Entwurfsingenieur alle Verkettungsmöglichkeiten kennen, um selbst einen Entwurfsplan zu finden. Jetzt schlägt ihm das Beratungssystem Entwurfswege vor, bietet ihm also die für sein Problem anzuwendenden Moduln in der richtigen Reihenfolge an. Das übersteigt wesentlich den Funktionsumfang von Menü-Techniken und HELP-Komponenten, deren Auskunft über die Verwendung der einzelnen Entwurfsmoduln die aktuelle Entwurfsaufgabe und den Bearbeitungsstand ja nicht berücksichtigt.

Von den vier im Abschn. 4. aufgeführten Bestandteilen des durch das Beratungssystem zu verarbeitenden Wissens ist nur das Wissen (W1) vom Entwurfssystem abhängig. Um die Tafel 1 zusammenzustellen, müssen die Zielstellungen und Einsatzbedingungen der Entwurfsmoduln bekannt sein. Ein Austausch der Entwurfssysteme macht lediglich den Austausch dieser als Wissensbasis verwendeten Tabelle notwendig.

(wird fortgesetzt)

R. Palm

Steuerung eines sensorgeführten Roboters unter Berücksichtigung eines unscharfen Reglerkonzepts

0. Einleitung

Unscharfe Regelungskonzepte (Fuzzy-Regler) kommen vorteilhaft dort zur Anwendung, wo eine hinreichende Beschreibung des zu regelnden Prozesses wegen dessen Komplexität bzw. nichtlinearen Charakters auf Schwierigkeiten stößt und daher einen erheblichen Aufwand erfordert. In [1] und [2] wurden Beispiele zur Regelung eines Wärmeaustauschers bzw. einer kleinen Dampfmaschine unter Anwendung linguistischer Produktionsregeln vom Typ „IF A THEN B ELSE C" vorgestellt. Anwendungen in der Roboterforschung sind in [3] bis [5] angegeben. Nichtlinearitäten und die Schwierigkeit der Einstellung von Reglerparametern eines konventionellen Reglers bei speziellen Bearbeitungsprozessen (z. B. Fräsen) führten zur Wahl eines Fuzzy-Reglerkonzepts [6] und [7]. Der im folgenden vorgeschlagene Fuzzy-Regler wird in der äußeren Regelschleife eines Roboters mit externer Sensorik (z. B. Abstands- oder Kraftsensorik) eingesetzt (Bild 1).

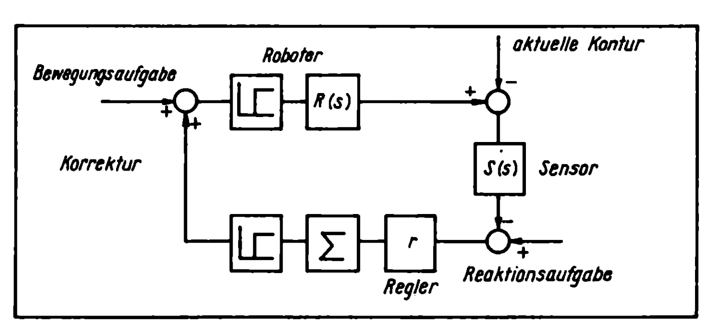


Bild 1. Externe Regelschleife eines sensorgeführten Roboters

1. Grundoperationen

Gegeben sei die Fuzzy-Menge Y durch die Menge geordneter Paare

$$Y = (\mu Y(x), x), \qquad \forall x \in X,$$

wobei die Zugehörigkeitsfunktion $\mu Y(x)$ den Grad der Zugehörigkeit von x in Y anzeigt und

$$\mu Y(x) \in [0, 1]$$

ist.
Es sei beispielsweise Y die Menge junger Personen, wobei die Bezeichnung „jung" für eine Person mit dem Alter x nur unscharf vergeben werden kann. So werde eine 30 Jahre alte Person mit dem Zugehörigkeitsgrad $\mu Y(30) = 0{,}6$ als „jung" bezeichnet, während ein 10jähriges Kind mit $\mu Y(10) \approx 1$ mit „jung" bezeichnet wird (Bild 2).

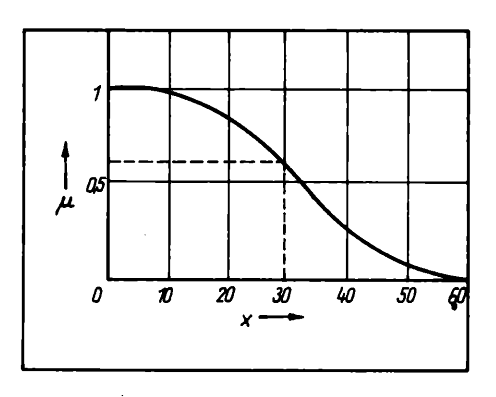


Bild 2. Zugehörigkeitsfunktion für „jung"

Für einen Fuzzy-Regler werden folgende Grundoperationen benutzt:

1. *Komplement* (Bild 3a)

$$\neg A = (1 - \mu A(x), x), \qquad \forall x \in X$$

2. *Vereinigung* (Bild 3b)

$$A \cup B = (\mu A(x) \cup \mu B(x), x), \qquad \forall x \in X,$$

mit

$$\begin{aligned}\mu A(x) \cup \mu B(x) &= \max(\mu A(x), \mu B(x)) \\ &= \mu A(x) \vee \mu B(x) \\ &= (\mu A(x) + \mu B(x) + |\mu A(x) - \mu B(x)|)/2\end{aligned}$$

3. *Durchschnitt* (Bild 3c)

$$A \cap B = (\mu A(x) \wedge \mu B(x), x), \qquad \forall x \in X,$$

mit

$$\begin{aligned}\mu A(x) \cap \mu B(x) &= \min(\mu A(x), \mu B(x)) \\ &= \mu A(x) \wedge \mu B(x) \\ &= (\mu A(x) + \mu B(x) - |\mu A(x) - \mu B(x)|)/2\end{aligned}$$

4. *Kartesisches Produkt*

$$A \times B = (\mu A(x) \cap \mu B(y), (x, y)), \qquad \forall x, y \in X \times Y$$

5. *Max-Min-Komposition*

$$R \circ S = \max(\mu R(x, z) \cap \mu S(z, y), (x, y))$$

mit

$$\max(\mu R(x, z) \cap \mu S(z, y)) = \max(\min(\mu R(x, z), \mu S(z, y)))\,.$$

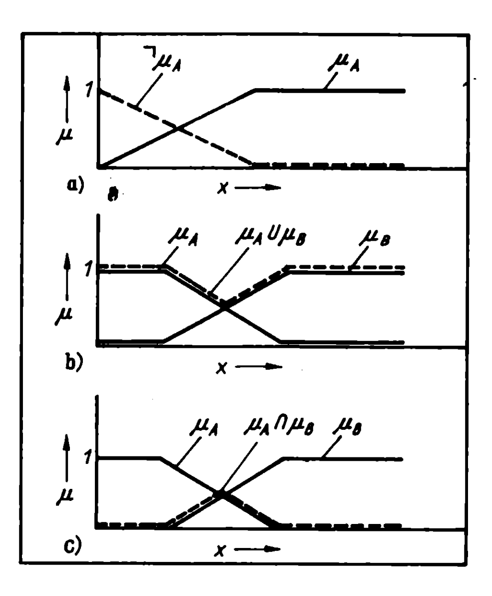


Bild 3. Basisoperationen

Dr.-Ing. RAINER PALM (46) studierte von 1966 bis 1969 an der Ingenieurschule für Elektrotechnik und Maschinenbau Berlin-Lichtenberg in der Fachrichtung Elektroanlagenprojektierung. Von 1969 bis 1970 Prüffeldingenieur im VEB Kombinat EAB, von 1970 bis 1981 Systemingenieur am Institut für Regelungstechnik Berlin. Von 1969 bis 1973 Fernstudium „Technische Kybernetik" an der TU Dresden. 1981 Promotion A. 1982 bis 1988 wissenschaftlicher Mitarbeiter am Zentralinstitut für Kybenetik und Informationsprozesse (ZKI) der AdW. Seit 1988 wissenschaftlicher Mitarbeiter am Institut für Automatisierung der AdW. 1988 Dissertation B eingereicht.

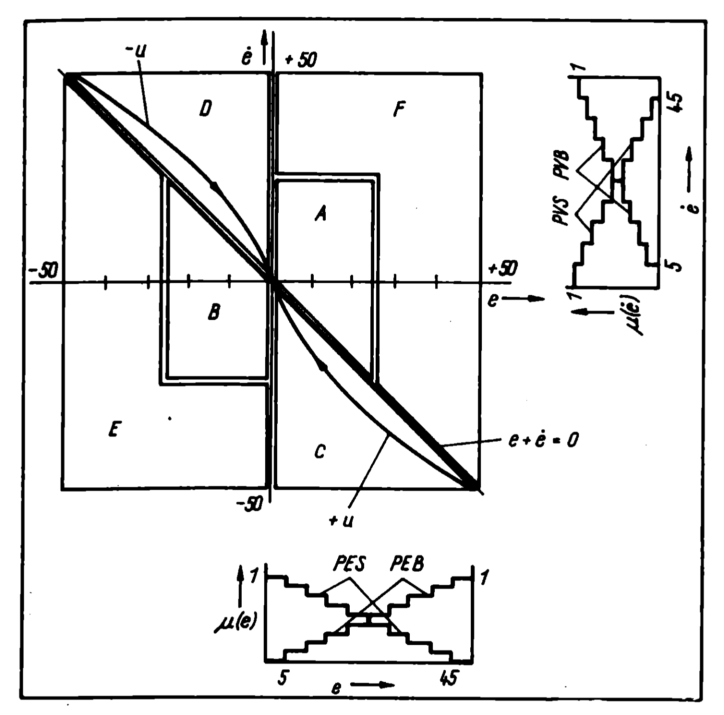


Bild 4. Phasenebene und Zugehörigkeitsfunktionen

Fuzzy-Regler werden gewöhnlich mit Hilfe von Fuzzy-Implikationen „IF A THEN B" realisiert [2]. Diese Produktionsregel entspricht dem kartesischen Produkt der Fuzzy-Mengen A und B.

$$A \rightarrow B \triangleq (\mu A(x) \cap \mu B(y), (x, y)), \quad \forall x, y \in X \times Y$$

Weiterhin gelten:

NOT Komplement $\neg$
OR Vereinigung $\cup$
AND Durchschnitt $\cap$.

2. Fuzzy-Reglerkonzept

Der im folgenden vorgestellte Regler wurde auf der Basis der Phasenebene eines Systems 2. Ordnung entworfen (Bild 4). Wie der naheliegende Vergleich zu einem Regler zeigt, der im sliding-mode arbeitet (siehe [9]), ist sein Anwendungsbereich aber nicht allein auf lineare Systeme beschränkt, was ihn zu einem robusten Regler hinsichtlich Systemparameterschwankungen macht. Es seien $\tilde{e} = x(t) - x(t)_{\text{soll}}$ der Fehler, $\dot{\tilde{e}} = \dot{x}(t) - \dot{x}(t)_{\text{soll}}$ die Fehlergeschwindigkeit und $\tilde{u}(t)$ sei die Steuergröße. Diese Größen werden so normiert, daß die ihnen zugeordneten Zugehörigkeitsfunktionen innerhalb eines gemeinsamen Standardintervalls gebildet werden können. Zwischen den normierten und den physikalischen Größen gelten die Zusammenhänge:

$$e = FE \cdot \tilde{e},$$
$$\dot{e} = FV \cdot \dot{\tilde{e}},$$
$$u = LF \cdot \tilde{u}.$$

Für den Fehler e, die Fehlergeschwindigkeit $\dot{e}$ und die Steuergröße u wurden die folgenden diskreten Fuzzy-Mengen gebildet:

positiv groß $e = PEB$; negativ groß $e = NEB$
positiv klein $e = PES$; negativ klein $e = NES$
positiv groß $\dot{e} = PVB$; negativ groß $\dot{e} = NVB$
positiv klein $\dot{e} = PVS$; negativ klein $\dot{e} = NVS$
positiv groß $u = PUB$; negativ groß $u = NUB$
positiv klein $u = PUS$; negativ klein $u = NUS$.

Die Zugehörigkeitsfunktionen wurden als lineare Funktionen innerhalb eines beliebig festgelegten Standardintervalls aus heuristischen Erwägungen entworfen und dann diskretisiert, um sie tabellarisch abspeichern zu können.
Mit Hilfe dieser linguistischen Variablen wird der Regler anhand der oben erwähnten Fuzzy-Implikationen aufgebaut. Die Phasenebene wird hierzu in unscharf gegeneinander abgegrenzte Gebiete A bis F eingeteilt, wobei analog zu einer suboptimalen Regelung [8] die Gerade $e + \dot{e} = 0$ zur Umschaltung des Vorzeichens von u dient. In diesem Sinne wurden folgende Regeln gewählt.

Gebiet		Steuergröße	
IF *F*	THEN	*NUB*	
IF *A* OR *D*	THEN	*NUS*	(1)
IF *B* OR *C*	THEN	*PUS*	
IF *E*	THEN	*PUB*	

Diese Gebiete sind wie folgt erklärt:

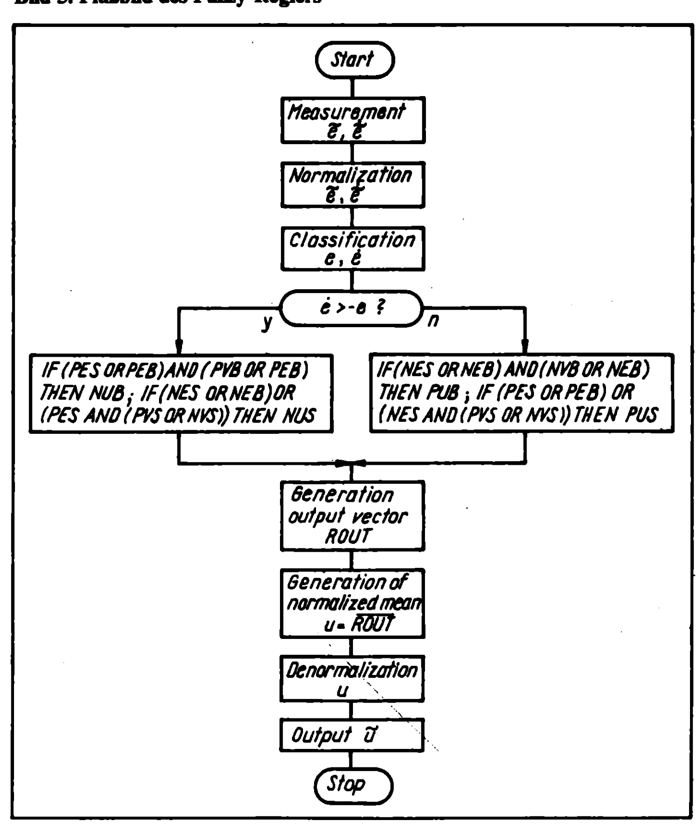
A: $(\dot{e} > -e)$ AND PES AND $(PVS$ OR $NVS)$
B: $(\dot{e} \leqq -e)$ AND NES AND $(PVS$ OR $NVS)$
C: $(\dot{e} \leqq -e)$ AND $(PES$ OR $PEB)$
D: $(\dot{e} > -e)$ AND $(NES$ OR $NEB)$ (2)
E: $(\dot{e} \leqq -e)$ AND $(NES$ OR $NEB)$ AND $(NVB$ OR $NEB)$
F: $(\dot{e} > -e)$ AND $(PES$ OR $PEB)$ AND $(PVB$ OR $PEB)$.

Eine Zusammenfassung von (1) und (2) ergibt

IF $\dot{e} > -e$ THEN
IF $(PES$ OR $PEB)$ AND $(PVB$ OR $PEB)$ THEN NUB
IF $(NES$ OR $NEB)$ OR $(PES$ AND $(PVS$ OR $NVS))$
THEN NUS (3)
IF $\dot{e} \leqq -e$ THEN
IF $(NES$ OR $NEB)$ AND $(NVB$ OR $NEB)$ THEN PUB
IF $(PES$ OR $PEB)$ OR $(NES$ AND $(PVS$ OR $NVS))$
THEN PUS.

Die Anwendung der Regel (3) führt zu einem Fuzzy-Vektor $ROUT$ für die normierte Steuergröße u, aus dem durch Mittelwertbildung und Denormierung die aufzubringende skalare Steuergröße $\tilde{u}$ berechnet wird. Bild 5 zeigt das entsprechende Flußbild des Fuzzy-Reglers.
Ein wesentliches Problem ist die Normierung der physikalischen Größen $\tilde{e}$, $\dot{\tilde{e}}$ und $\tilde{u}$. Die entsprechenden Normierungsfaktoren FE, FV und LF sind als Regelparameter zu behandeln. Die Anpassung des Reglers an die i. allg. nichtlineare Regelstrecke erfolgt im we-

Bild 5. Flußbild des Fuzzy-Reglers



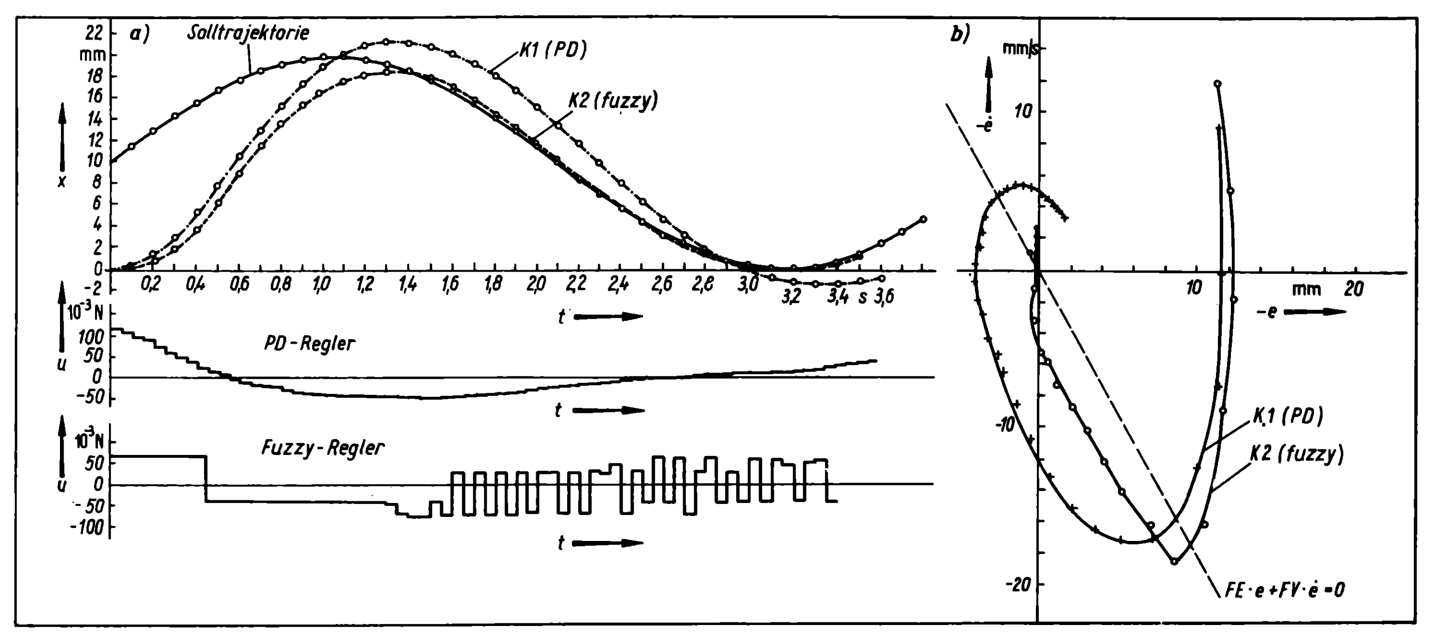


Bild 6. Simulationsergebnisse
a) Annäherung an die Solltrajektorie in der Zeitebene; b) Darstellung in der Phasenebene

sentlichen über die Dimensionierung dieser Parameter. Die Normierung erfolgt so, daß im stationären Fall die Erwartungswerte von e und $\dot{e}$ in der Mitte der entsprechenden Intervalle liegen, innerhalb derer die Zugehörigkeitsfunktionen für e und $\dot{e}$ definiert sind. Die Wahl des Denormierungsfaktors erfolgt so, daß die Gleitbedingung

$$s \cdot \dot{s} < 0$$

mit

$$s = c \cdot \tilde{e} + \dot{\tilde{e}}, \qquad c > 0$$

erfüllt ist (siehe [9]).

3. Simulationen und praktische Resultate

Der Fuzzy-Regler wurde an einem System 2. Ordnung mit trockener Reibung und einer sinusförmigen Solltrajektorie getestet und mit einem PD-Regler verglichen, wobei der Fuzzy-Regler zwar einen höheren Implementierungsaufwand verlangt, sich aber leicht in Echtzeit abarbeiten läßt, wenn die Zugehörigkeitsfunktionen tabellarisch abgespeichert sind. Der PD-Regler produziert wie erwartet periodischer Fehler, während der Fuzzy-Regler diese Fehler sehr gering hält (Bild 6a). Eine Begleiterscheinung dieser guten Folgequalität ist allerdings eine hohe Aktivität der Steuergröße und damit verbunden ein schnelles Alternieren von $\dot{e}$ in der Nähe des Nullpunktes der Phasenebene (Bild 6b).
Für diese Simulation gelten die folgenden Vereinbarungen:

$$m \cdot \ddot{x} + Kv \cdot \dot{x} + Kf \cdot \operatorname{sgn} \dot{x} = u$$

$$m = 1\ \text{kg}; \quad Kv = 0{,}2\ \text{kg/s}; \quad Kf = 7\ \text{kg mm/s}^2$$

$$x_{\text{soll}} = 10 \cdot (1 + \sin 1{,}5 \cdot t/\text{s})\ \text{mm}$$

$$P = 3\ \text{kg/s}^2; \quad D = 7\ \text{kg/s}$$

$$LF = 0{,}5\ \text{s}^2/\text{kg} \cdot \text{mm}; \quad FE = 180 \cdot 1/\text{mm}; \quad FV = 100\ \text{s/mm}$$

Abtastzeit $T = 0{,}05$ s.

Für den praktischen Funktionsnachweis wurde ein an einem Robotereffektor befindlicher taktiler Sensor zum Verfolgen einer rampenförmigen Kontur benutzt. Hervorzuheben ist, daß die Systemstruktur derjenigen im Bild 1 entspricht und daß der Ausgang des Reglers eine Positionskorrektur darstellt, die ihrerseits zu einer Korrektursumme integriert wird (siehe [10]). Daher gelten die obigen Aussagen zum sliding-mode hier nicht. Bild 7a zeigt die Ergebnisse mit einem P-Regler für die Verfahrgeschwindigkeit des Arms, was einem I-Regler für die Effektorposition entspricht. Das Verfolgen der

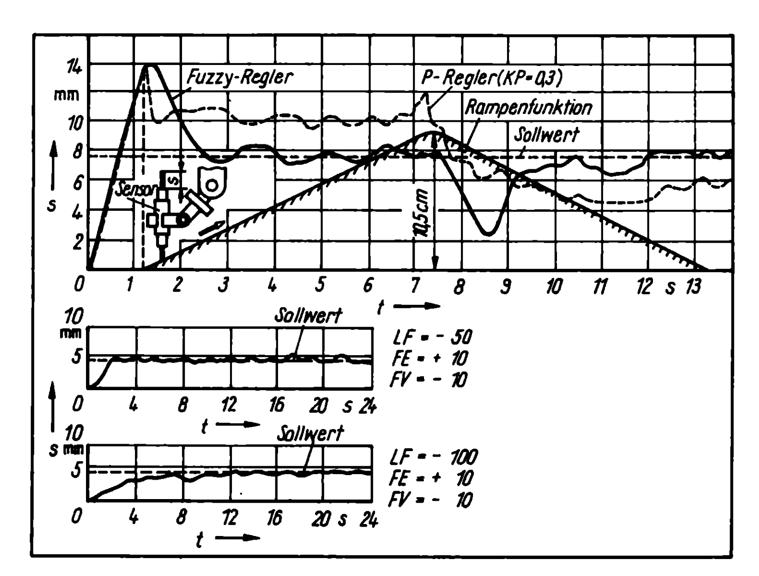


Bild 7. Experimentelle Ergebnisse
LF Normierung $\tilde{u}$; *FE* Normierung $\tilde{e}$; *FV* Normierung $\dot{\tilde{e}}$

Rampe weist wie erwartet eine bleibende Regelabweichung auf, die gegebenenfalls durch einen weiteren Integrationsterm beseitigt werden könnte. Allerdings ergäben sich durch den nunmehr entstandenen Doppelintegrator bei diesem konventionellen Regler Stabilitätsprobleme. Ein solcher weiterer Integrationsterm wurde dagegen problemlos beim Fuzzy-Regler eingesetzt. Dieses führt erwartungsgemäß zum Verschwinden der bleibenden Regelabweichung, wobei jedoch beim Überschreiten des Konturgipfels eine höhere Aktivität zu verzeichnen ist.
Der Annäherungsvorgang des Robotereffektors unter Zuhilfenahme eines Kraftsensors ist im Bild 7b mit unterschiedlichen Denormierungsfaktoren *LF* dargestellt, wobei hier ohne einen zusätzlichen Integrator gearbeitet wurde. Verglichen mit einem PD-Regler spielt *FE*/*LF* die Rolle des P-Verstärkungsfaktors und *FV*/*LF* die des D-Verstärkungsfaktors. Eine Erhöhung von *LF* macht die Reaktion des geregelten Systems somit träger.

Zusammenfassung

Die Regelung sensorgeführter Roboter erfordert wegen des nichtlinearen Charakters gewisser Reaktionsprozesse zwischen Robotereffektor und Objekt die Nutzung robuster Regelungsalgorithmen, die leicht implementiert werden können. Es wird ein Fuzzy-Regler vorgestellt, der auf der Basis der Phasenebene für Systeme 2. Ord-

nung entworfen wird. Hierzu wird die Phasenebene zunächst in 6 unscharfe Regionen eingeteilt. Für den Fehler, die Fehlergeschwindigkeit und die Steuergröße werden 12 Fuzzy-Mengen mit den korrespondierenden Zugehörigkeitsfunktionen gebildet. Fuzzy-Produktionsregeln führen zu einem unscharfen Steuervektor, aus dem durch Mittelwertbildung ein scharfer Steuerwert gebildet wird.
In Zukunft sind weitere Untersuchungen zur Anwendung von Fuzzy-Konzepten bei Robotern insbesondere auf dem Gebiet der sensorgeführten Montage geplant. Entsprechende Simulationsergebnisse liegen bereits vor. An einer Realisierung mit Hilfe von Kraftsensoren wird gegenwärtig gearbeitet.

Literatur

[1] Østergaard, J. J.: Fuzzy Logic Control of a Heat Exchanger Process. In: Fuzzy Automata and Decision Processes. Hrsg.: Gupta, M. M. Amsterdam/New York/Oxford: North-Holland 1977, S. 285 bis 320.

[2] King, P. J.; Mamdani, E. H.: The Application of Fuzzy Control Systems to Industrial Processes. In: Fuzzy Automata and Decision Processes. Hrsg.: Gupta, M. M. Amsterdam/New York/Oxford: North-Holland 1977, S. 321—330.

[3] Saridis, G. N.; Stephanou, H. E.: Fuzzy-Decision-Making in Prosthetic Devices. In: Fuzzy-Automata and Decision Processes. Hrsg.: Gupta, M. M. Amsterdam/New York/Oxford: North-Holland 1977, S. 387—402.

[4] Hirota, K.; Arai, Y.; Pedrycz, W.: Robot Control Based on Membership and Vagueness. In: Approximate Reasoning in Expert Systems. Hrsg.: Gupta, M. M.; Kandel, A.; Bandler, W.; Kizka, J. B. Amsterdam/New York/Oxford: Elsevier Science Publishers B.V. (North-Holland) 1985, S. 622—635.

[5] Lakov, D.: Adaptive Robot under Fuzzy Control. Fuzzy Sets and Systems 17 (1985), S. 1—8.

[6] Palm, R.: Design of a Fuzzy Controller for a Sensor Guided Robot Manipulator. 2nd Symposium on Foundation of Kinematics, Dynamics and Control of Manipulation-Robots. Schwerin/DDR 1986.

[7] Zecha, M.; Palm, R.: Hierarchical Processing of Tactile and Range Sensor Data for Robot Control. IFAC/IFIP/IMACS International Symposium on Theory of Robots, 3. bis 5. Dezember 1986, Wien, S. 451—455.

[8] Föllinger, O.: Nichtlineare Regelungen II. Berlin: Akademie-Verlag 1981.

[9] Choi, S.-B.; Jayasuriya, S.: A Sliding Mode Controller. Incorporating Matching Conditions Applied to Manipulators. 10. IFAC-Weltkongreß, München 1987. Proceedings S. 290—295.

[10] Palm, R.: Probleme der Steuerung von Industrierobotern mit externer Sensorik. msr, Berlin 29 (1986) 9, S. 402—407.

msr 8957

A. Iwainsky

Rechnerunterstützte grafische Dokumentation in CAD-Prozessen

0. Einleitung

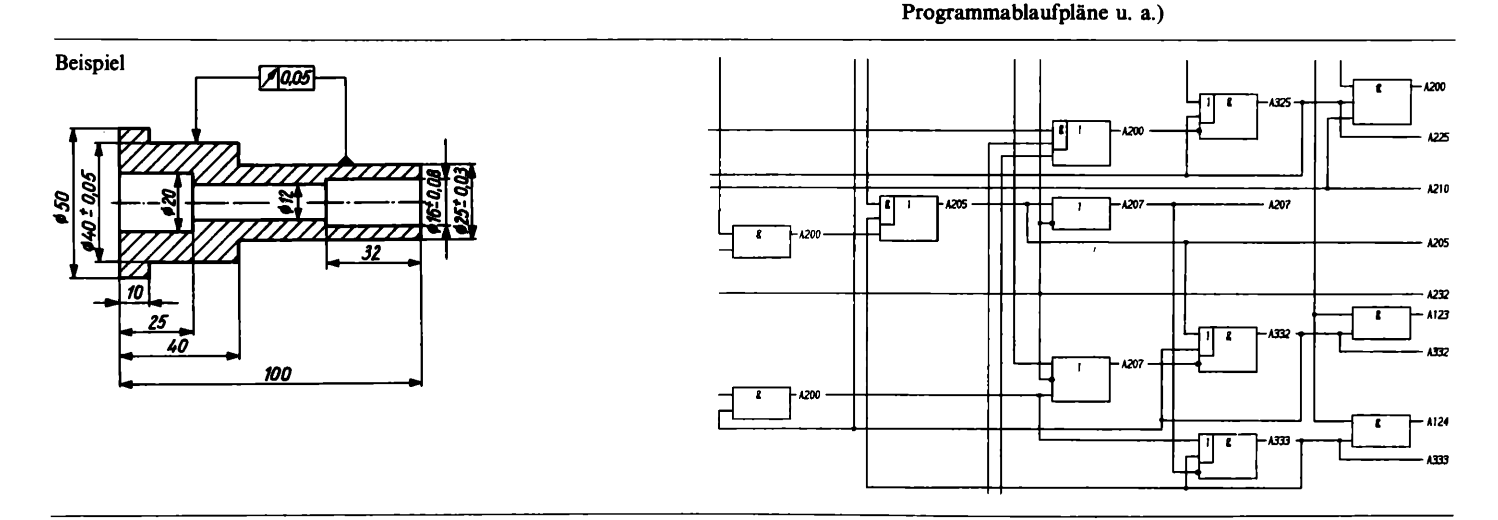
Mit dem Vordringen der Rechnerunterstützung in den kreativen, funktionsorientierten Entwurfs- bzw. Konstruktionsbereich erhält die Erzeugung grafischer Dokumentationen einen neuen Stellenwert. Im Interesse der Integration verschiedener CAD-Prozesse sowie der weiteren Aufwandsreduktion besteht eine große Herausforderung darin, die grafische Darstellung möglichst weitgehend automatisiert aus demjenigen Produkt-, Prozeß- oder Projektmodell abzuleiten, das sie lediglich illustrieren soll. Der Beitrag konzentriert sich auf zwei Grundtypen grafischer Dokumentationen, bei denen diese Herausforderung besonders deutlich wird. Dabei handelt es sich um die klassische Werkstattzeichnung sowie um netzartige grafische Schemata. Es wird u. a. dargelegt, welche Anforderungen an rechnerinterne 3D-Modelle zu stellen sind, um bestimmte Teilprozesse der Erstellung von Werkstattzeichnungen automatisieren und damit den heute noch relativ hohen interaktiven Aufwand reduzieren zu können. Im Fall netzartiger Schemata ist aufgrund ihrer strukturorientierten Informationspräsentation prinzipiell die Möglichkeit einer sehr weitgehenden Automatisierung ihrer Erzeugung gegeben. Als konkretes Beispiel für die Realisierung dieser Möglichkeit wird eine Entwicklung vorgestellt und diskutiert, die zur vollautomatischen Erzeugung von Logikplänen aus der Beschreibung der Funktion industrieller Steuerungen geführt hat. Abschließend wird die Fortführung dieser Entwicklung angedeutet.

1. Überholte Probleme oder neue Herausforderung?

Zwar war eine frühere Entwicklungsphase von CAD mit der rechnerunterstützten Dokumentation eng verbunden, doch haben inzwischen der Slogan von der papierlosen Fabrik und andere Aspekte zu einer gewissen Unterbewertung der entsprechenden Probleme geführt. Verbreitet ist z. B. die vielleicht nicht immer vollständig bewußte Meinung, daß hübsche bunte Bilder auf Displays nüchterne Schwarz-Weiß-Darstellungen auf Papier nach und nach verdrängen würden und daß u. a. deshalb die Erzeugung grafischer Dokumentationen eine Aufgabe ist, die der Vergangenheit angehört. Hinter solchen Vorstellungen stehen jedoch einige Mißverständnisse. Zunächst sei bemerkt, daß der Prozeß der Dokumentation nicht unbedingt zu bedrucktem bzw. mit Grafiken versehenem Papier führen muß. Dies sollte in einer Zeit der elektronischen Informationsverarbeitung und -übertragung eigentlich selbstverständlich sein, aber häufig assoziiert man mit dem Begriff „Dokumentation" die Vorstellung von Papier. Weiterhin gerät oft in Vergessenheit, daß konventionelle technische Zeichnungen reale Objekte meist kompakter, umfassender und präziser repräsentieren als hübsche bunte Computergrafiken. Damit soll nicht gegen die spezifische Bedeutung anschaulicher farbiger Darstellungen (z. B. bei der interaktiven Generierung von 3D-Modellen) Stellung bezogen werden. Im Rahmen grafischer Dokumentationen kann man aber in weiten Anwendungsbereichen auf sie verzichten, was bei konventionellen (d. h. auf fundierten Konventionen beruhenden) technischen Zeichnungen nicht der Fall ist. Es geht also nicht darum, durch Rechnerunterstützung produktionsvorgelagerter Prozesse grafische Dokumentationen in Form und Inhalt weitgehend zu verändern oder gar völlig abzuschaffen. Die Herausforderung an die Rechnerunterstützung besteht vielmehr darin, konventionelle grafische Dokumentationen mit unkonventionellen Methoden und Hilfsmitteln zu erzeugen. Das Ziel ist vor allem eine drastische Senkung des Aufwandes für die rechnerunterstützte Dokumentation, also eine möglichst weitreichende Automatisierung dieses Prozesses. Es läßt sich nur erreichen, wenn die Kluft zwischen rechnerunterstützter, über die grafische Darstellung hinausgehender Modellierung einerseits und rechnerunterstützter grafischer Dokumentation andererseits überwunden wird.
Besonders deutlich wird diese Herausforderung bei zwei Grundtypen von grafischen Dokumentationen, die in allen hochentwickelten Volkswirtschaften in großen Mengen zu erzeugen sind. Es handelt sich dabei um klassische Werkstattzeichnungen, also um grafische Darstellungen, die dreidimensionale Objekte maßstabsgerecht repräsentieren, sowie um netzartige grafische Schemata (Tafel 1).

Prof. Dr. sc. nat. Alfred Iwainsky (40) studierte von 1967 bis 1972 Physik an der Humboldt-Universität zu Berlin. 1972 bis 1978 wissenschaftlicher Mitarbeiter am heutigen Institut für Kosmosforschung der AdW der DDR. 1976 Promotion A (Extrapolation und Identifikation von Zufallsprozessen). Ab 1979 wissenschaftlicher Mitarbeiter im Zentralinstitut für Kybernetik und Informationsprozesse (ZKI) der AdW der DDR. 1982 Berufung zum Leiter der Abteilung „Prozeßautomatisierung". 1986 Promotion B (Rechnergestütztes Layout von Verbindungsstrukturen in industriellen Umgebungen). 1987 Berufung zum wissenschaftlichen Bereichsleiter und Ernennung zum Professor an der AdW der DDR.

Tafel 1. Charakterisierung zweier Grundtypen grafischer Dokumentationen

	Werkstattzeichnung	netzartige Schemata
wichtigste Bestandteile	— Hauptkomponenten: Konturen von Ansichten, Schnitten und Detaildarstellungen — Bemaßungen — Toleranzangaben — technologische Angaben (z. B. zur Oberflächengüte) — Schraffuren von Flächenbereichen — sonstige grafische Hilfselemente wie Symmetrie- oder Biegelinien — Schriftfeld und Zeichnungsrahmen — sonstige Zeichenketten	— Symbole bzw. Sinnbilder — Verbindungslinien zwischen bestimmten Anschlußstellen von Symbolen — Zeichenketten — grafische Hilfselemente wie Umrandungen von Funktionsblöcken — Schriftfeld
Kurzcharakterisierung	Konturen von Ansichten, Schnitten und Detaildarstellungen sind maßstabsgerecht; Geometrie außerdem durch alphanumerische Angaben (Maße, Toleranzen) spezifiziert; Anordnung der Hauptkomponenten erfolgt so, daß geometrische Beziehungen zwischen ihnen leicht hergestellt werden können und die Rekonstruktion eines (mentalen) 3D-Modells unterstützt wird.	Maßstabstreue spielt keinerlei Rolle. Der Informationsgehalt liegt im wesentlichen in der Struktur der Grafik, d. h. in der Darstellung von Sinnbildern bzw. Symbolen und der Relationen zwischen ihnen (meist durch Verbindungslinien zwischen Symbol-Pins ausgedrückt); es gibt eine ganze Reihe einzelner Typen netzartiger Schemata, deren grafische Darstellung Konventionen zu genügen hat (Logikpläne, Stromlaufpläne, technologische Schemata, Programmablaufpläne u. a.)
Beispiel		

Im Fall der Werkstattzeichnung geht es darum, an die Stelle der unmittelbaren Zeichnungserstellung mit den konventionellen Hilfsmitteln interaktiver 2D-Systeme die weitgehend automatisierte Ableitung der grafischen Dokumentation aus dem entsprechenden 3D-Modell zu setzen. Derzeit kann der Grad der Integration von rechnerunterstützter 3D-Modellierung und rechnerunterstützter Zeichnungserstellung bei weitem noch nicht befriedigen. Wenn Konstrukteure davon sprechen, daß die Zeichnungserstellung mit einem für diesen Zweck gut zugeschnittenen 2D-System meist schneller durchgeführt werden kann als beim „Umweg" über die 3D-Modellierung, so ist dies ein Ausdruck dafür. Dieses Beispiel zeigt auch, in welch hohem Maß seitens der Nutzer entsprechender Systeme von der Rechnerunterstützung Rationalisierungseffekte im jeweils eigenen Arbeitsbereich erwartet werden. Mehraufwand zugunsten anderer, sie nicht direkt betreffender Bereiche wird von ihnen i. allg. nicht akzeptiert. Die 3D-Modellierung erscheint ihnen dann eben als Umweg, als Ballast, selbst wenn andere betriebliche Bereiche erhebliche Effekte durch die Weiterverwendung der erzeugten 3D-Modelle erzielen können.

Im Falle netzartiger Schemata liegt eine ähnliche Herausforderung an die Rechnerstützung ihrer Erzeugung vor wie bei der Werkstattzeichnung. Sie ist allerdings bisher selten erkannt und adäquat berücksichtigt worden. Die besonders hohen, noch weitgehend unerschlossenen Rationalisierungsreserven resultieren aus der spezifischen Art der Informationsrepräsentation durch netzartige Schemata. Wesentlich für den Informationsgehalt eines netzartigen Schemas sind einerseits die Auswahl gewisser Symbole aus einem entsprechenden Konventionen genügenden fachspezifischen Symbolvorrat und andererseits die Beziehungen zwischen diesen Symbolen, die meist durch Verbindungslinien zwischen bestimmten Anschlußstellen der Symbole ausgedrückt werden (Tafel 1). Bei der Erzeugung netzartiger Schemata mit konventionellen interaktiven 2D-Systemen dient nur ein geringer Teil der Arbeit des Entwurfes der Spezifikation dieser wesentlichen Charakteristika von Schemata. Hohen Aufwand bereitet dagegen die ständige Konfrontation mit Problemen der Gestaltung ihres Erscheinungsbildes. Derartige Layout-Probleme können dem Entwerfer durch Rechnerunterstützung prinzipiell abgenommen werden. Die Verbindung der generativen Computergrafik mit der Layout-Optimierung ermöglicht im Fall der netzartigen Schemata nicht nur das Erreichen extrem hoher Produktivitätssteigerungen, sondern auch den direkten Anschluß der rechnergestützten Schemata-Erzeugung an nichtgrafische, funktionsorientierte rechnerinterne Modelle.

Der Umfang von grafischen Dokumentationen in der industriellen Praxis sei durch zwei Zahlen verdeutlicht:

— Bereits in mittelgroßen Unternehmen enthalten die Archive Zehntausende von Zeichnungen (s. z. B. [1]).
— Die grafische Dokumentation programmierbarer Steuerungen kann schon für eine einzige Anlage (z. B. eine Feinwalzstraße) ungefähr 1000 A3-Blätter mit Logikplänen umfassen [2].

Die traditionelle Verwendung grafischer Dokumentationen in der Industrie brachte mit dem Vordringen der Rechnerstützung in Bereiche außerhalb der reinen Zeichnungserstellung den dringenden Bedarf mit sich, manuell erstellte Zeichnungen aufwandsarm in CAD-Systeme eingeben und interpretieren zu können (s. z. B. [1], [3] und [4]).

Die Notwendigkeit der massenweisen Erstellung grafischer Dokumentationen und die relativ einfachen Möglichkeiten der Unterstützung dieses Prozesses durch digitalgrafische Systeme führte

Tafel 2. Produktivitätssteigerungen durch Rechnerunterstützung der Zeichnungserstellung

Nr.	Anwendungsbereich	Bemerkungen	Beschleunigungsfaktor	Quelle
1	Maschinenbau	Werkstattzeichnungen von Blechteilen Layout-Darstellungen Logikpläne	3,7 1,7 ... 2,2 4,5 ... 5,0	[5] mit Bezug auf [6]
2		Zeichnungserstellung im Bereich der „allgemeinen Konstruktion mit allgemeinen CAD-Systemen"	5	[5] mit Bezug auf [7]
3	Maschinenbau/Elektrotechnik	Schaltanlagen, Transformatoren, Schaltpläne; Hilfsmittel: 8 CAD-Systeme, 14 Arbeitsplätze; Off-line-Kopplung zu Großrechnern	2 ... 3	[5] mit Bezug auf [7]
4	Elektrotechnik	gedruckte Schaltungen mittels schlüsselfertigem CAD-System	4 ... 5	[5] mit Bezug auf auf [7]
5	Elektrotechnik Elektronik	grafische Schemata	4,2	[8]
6	Maschinenbau	Anwendung von PROREN1 auf zwei Zeichnungen, die zum Vergleich auch konventionell erstellt wurden (die beiden Zeichnungen sind in [5] abgedruckt); Vergleich erfolgte bei *Fichtel & Sachs*	2	[5], S. 138ff.
7		wie Fall 6, aber mit DETAIL2	6	[5], S. 138ff.
8	Anlagenbau	Hydraulikblöcke mit 16, 20 bzw. 40 Ventilen unter Nutzung von EUCLID; Zeichnungserstellung: Zusammenstellungszeichnung:	 4,7; 4 bzw. 4 3; 2,4 bzw. 2,25	[10]
9	Elektrotechnik Elektronik	vollautomatische Erstellung von Logikplänen aus einer Fachsprache für speicherprogrammierbare Steuerungen (SPS)	>60	[9], s. a. Abschn. 3.

bereits in einer frühen Phase der CAD-Entwicklung zu einem routinemäßigen Einsatz rechnerunterstützter Systeme für die Zeichnungserstellung. Dementsprechend konnte schon frühzeitig über Einsatzerfahrungen, insbesondere über Produktivitätssteigerungen berichtet werden. Tafel 2 gibt einen Eindruck davon. Die mitgeteilten Werte sind teilweise Durchschnittswerte, die sich aus längerfristigen Einsatzerfahrungen einer ganzen Reihe befragter Betriebe ergeben haben (z. B. im Fall 5), z. T. handelt es sich um genaue Analysen des Einsatzes eines einzelnen Systems (z. B. in den Fällen 6 bis 9). In den Fällen 6 und 7 wurden nur je zwei verschiedene Werkstattzeichnungen manuell bzw. rechnergestützt erstellt, diese sind in [5] abgedruckt.
In Tafel 2 sind die Fälle 7 und 9 die einizigen Beispiele, bei denen der Herausforderung nach einer weitgehenden Automatisierung der Erzeugung grafischer Dokumentationen aus funktionsorientierten rechnerinternen Modellen bereits entsprochen wurde.

Tafel 3. Typische Arbeitsteilung zwischen Mensch und System bei der interaktiven Zeichnungserstellung

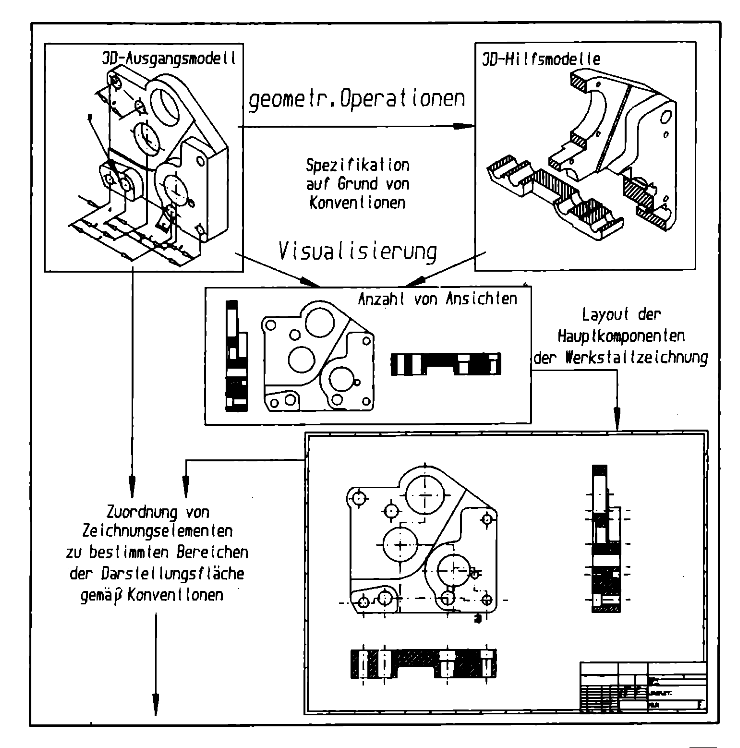
Mensch	System
— Festlegung der erforderlichen Ansichten, Schnitte, Detaildarstellungen; Spezifikation notwendiger geometrischer Operationen	— Durchführung geometrischer Operationen (Schnittbildung, Transformationen); Lösung des Verdeckungsproblems, Generierung der Konturzüge
— Auswahl der zu bemaßenden Positionen und der Objekte, die mit technologischen Angaben zu versehen sind	— Holen entsprechender Daten aus 3D-Modell sowie Bereitstellung von Symbolen
— Aktivitäten zur Festlegung des Layouts der Bemaßungen und technologischen Angaben	— physische Generierung der Bemaßungen und technologischen Angaben
— Festlegung zu schraffierender Flächenbereiche und der Schraffurart	— Generierung von Schraffuren
— Sonstiges (wie Festlegung von Linientypen, Durchführung von Korrekturen, Eintragen ins Schriftfeld)	— Automatische Erzeugung eines Zeichnungsrahmens und des Schriftfeldes — Bereitstellung von Daten für das Schriftfeld

2. Ableitung von Werkstattzeichnungen aus rechnerinternen 3D-Modellen

Mit dem Übergang von 2D- zu 3D-Systemen in weiten CAD-Anwendungsbereichen wird ein entscheidender Schritt in Richtung der Integration von Entwurf und Dokumentation vollzogen. Bei der rechnergestützten Zeichungserstellung mit Hilfe von 3D-Systemen werden die Konturen von Ansichten, Schnitten, Detaildarstellungen nicht mehr losgelöst voneinander erzeugt, sondern weitgehend automatisiert aus einem rechnerinternen 3D-Modell des darzustellenden Objektes abgeleitet. Der Systemnutzer hat lediglich festzulegen, welche Ansichten, Schnitte, Detaildarstellungen und welche entsprechenden internen geometrischen Operationen (wie Schnittbildungen) sowie Modellvisualisierungen zu veranlassen sind (Tafel 3 und Bild 1). Abgesehen von CAD-Lösungen für bestimmte Teileklassen (z. B. [11] und [12]) erfordert aber die rechnerunterstützte Zeichnungserstellung trotz dieses hohen Automatisierungsgrades bei der Erzeugung der Hauptbestandteile von Werkstattzeichnungen insgesamt einen erheblichen interaktiven Aufwand. Sie kann u. U. etwa genauso lang dauern wie die interaktive 3D-Modellierung. Weiterhin ist es möglich, daß der Zeitbedarf für die 3D-Modellierung und die Ableitung der Werkstattzeichnung zusammen in der gleichen Größenordnung liegt wie derjenige für die konventionelle Zeichnungserstellung. Wenn man nicht mehr mit 3D-Modellen vorhat als die Erzeugung entsprechender Werkstattzeichnungen, ist in solchen Fällen die Redeweise vom „Umweg über das 3D-Modell" nur allzu verständlich!
Der hohe Aufwand, der für die rechnerunterstützte Zeichnungserstellung im Rahmen der Nutzung von 3D-Systemen heute meist noch nötig ist, kann zwei Hauptgründe haben:

1. Zwischen 3D-Modellierungen und rechnergestützter Zeichnungserstellung besteht ein Bruch, der die Möglichkeiten der Übernahme von Informationen aus 3D-Modellen zur Verwendung im Prozeß der Zeichnungserstellung empfindlich einschränkt, und/oder
2. die Gestaltung des Erscheinungsbildes von Werkstattzeichnungen (Layout von Bemaßungen, Festlegung von Linientypen u. a.) ist schwerfällig und zeitraubend.

Der 1. Hauptgrund ist bei weitem der kritischere, weil er nicht nur die unmittelbare Zeichnungserstellung, sondern den ganzen Prozeß der rechnerunterstützten Konstruktion betrifft.



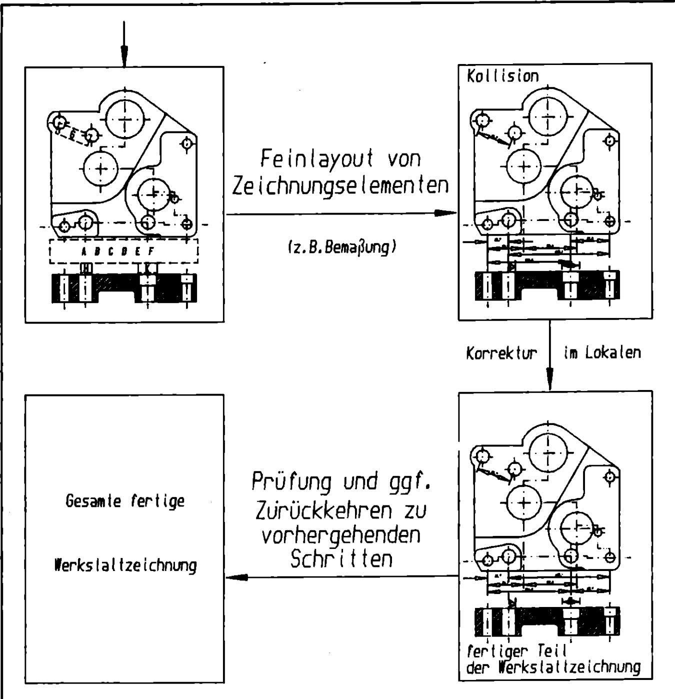


Bild 1. Schematische Darstellung der Ableitung von Werkstattzeichnungen aus 3D-Modellen

Tafel 3 zeigt die derzeit typische Arbeitsteilung zwischen Mensch und 3D-System bei der Ableitung von Werkstattzeichnungen aus 3D-Modellen. Man erkennt, daß der Mensch in entscheidenden Phasen des Gesamtprozesses aktiv werden muß. Nur ein relativ geringer Teil dieses Aufwandes steht mit kreativen, konstruktionsspezifischen Leistungen im direkten Zusammenhang. Der größte Teil betrifft lästige Aktivitäten zur norm- bzw. standardgerechten Gestaltung des gesamten Layouts einer Werkstattzeichnung, deren effektive Abwicklung aber durchaus nicht trivial ist.

Im Bild 1 ist die Ableitung von Werkstattzeichnungen aus 3D-Modellen in die wichtigsten Einzelprozesse zerlegt. Außerdem sind Ausgangsmodell und Hilfs- bzw. Zwischenmodelle anhand eines realen Werkstückes symbolisiert. Der o. g. 1. Hauptgrund für den z. Z. i. allg. noch niedrigen Automatisierungsgrad bei der Zeichnungserstellung wird im oberen Teil von Bild 1 angedeutet: Nur wenn im 3D-Modell diejenigen Informationen effektiv zugänglich sind, die für die Werkstattzeichnung benötigt werden, braucht der Systemnutzer das, woran er bei der Modellerzeugung zumindest gedacht hat, nicht erst im Rahmen der Zeichnungserstellung einzugeben. Dies betrifft u. a. die

— Verwendung funktional-technologischer Elemente im 3D-Ausgangsmodell und deren funktionsorientierte, relative Positionierung zueinander
— Integration von Toleranzangaben in das Ausgangsmodell
— Beschreibung der Geometrie sowohl des Ausgangsmodells als auch abgeleiteter 3D-Hilfsmodelle (z. B. Schnitte) in einer solchen Weise, daß in den entsprechenden Abbildern, also nach Ausführung von Visualisierungsprozessen keine Korrekturen (z. B. keine Überführungen von Polygonzügen in Kreisbögen) erforderlich sind
— Einbeziehung von Oberflächen- und Werkstoffangaben ins Ausgangsmodell
— automatische Kennzeichnung von Schnittflächen (auch von nicht zusammenhängenden) in aus dem Ausgangsmodell abgeleiteten 3D-Hilfsmodellen
— Integration von Angaben ins Ausgangsmodell, aus denen sich ergibt, welche Ansichten, Schnitte und Detaildarstellungen in die Werkstattzeichnung einzubeziehen sind.

Diese Liste zeigt, daß die Anforderungen an rechnerinterne 3D-Modelle seitens der Zeichnungserstellung zum großen Teil gleichzeitig auch Anforderungen anderer rechnerunterstützter Prozesse sind, die ebenfalls im Interesse der weiteren Rationalisierung der Produktionsvorbereitung einen möglichst effektiven Zugriff auf konstruktiv-technologisch möglichst vollständige 3D-Modelle benötigen. Dies gilt z. B. für den Anschluß der NC-Programmierung, die durch Berücksichtigung der im ersten bis vierten Anstrich erhobenen Forderungen ebenfalls weitgehend automatisiert durchgeführt werden kann. Diese weitgehende Übereinstimmung von Anforderungen seitens eines klassischen Prozesses der Produktionsvorbereitung (der Zeichnungserstellung) mit den aktuellen Herausforderungen an die Integration von CAD und CAM kann nicht überraschen, muß doch eine brauchbare Werkstattzeichnung gerade **fertigungsgerecht** sein!

Der gegenwärtig meist noch hohe Aufwand, der für die rechnerunterstützte Zeichnungserstellung im Rahmen von 3D-Systemen erforderlich ist, resultiert u. a. daraus, daß die in obigen Anstrichen aufgeführten Informationen im 3D-Modell entweder gar nicht zur Verfügung stehen oder das Subsystem zur Zeichnungserstellung nicht auf sie zugreift. Dies führt dazu, daß eine ganze Reihe der im Bild 1 dargestellten Übergänge zwischen verschiedenen Modell- bzw. Abbildrepräsentationen durch den Menschen veranlaßt bzw. spezifiziert werden müssen (z. B. auf der Basis von Konventionen und entsprechender Regelwerke) oder daß bestimmte Informationsflüsse rechnerintern gar nicht realisierbar sind. Letzteres gilt z. B. für die Extraktion von Angaben aus dem 3D-Modell darüber, was zu bemaßen ist.

Der o. g. 2. Hauptgrund tritt im mittleren und unteren Teil von Bild 1 in Erscheinung. Selbst wenn alle Informationen, die eine vollautomatische Zeichnungserstellung benötigt, im 3D-Modell auch vorhanden sind, verbleibt ein erheblicher Aufwand für die interaktive Erzeugung einer norm- bzw. standardgerechten Präsentation dieser Informationen in der Werkstattzeichnung. Im Bild 1 ist an einem typischen Beispiel lediglich ein Teil entsprechender Probleme verdeutlicht: Unterhalb der Hauptansicht des Werkstückes ist aus funktionalen Gründen (s. Bild 1, oben, Ausgangsmodell) eine ganze Reihe paralleler Maße anzubringen. Ein erster Versuch des Layouts dieser Maße führt zu einer Kollision mit einer der beiden Schnittdarstellungen in der Werkstattzeichnung. Eine Korrektur ist im lokalen Bereich möglich (andere Anordnung innerhalb der Maßgruppe). Wäre dies nicht der Fall, ergäbe sich die Notwendigkeit von Änderungen in der Anordnung der Hauptansicht und der Schnitte oder sogar im vorgesehenen Zeichnungsformat.

Die hier dargestellte Situation legt nahe, Hilfsmittel für die weitere Rationalisierung der Zeichnungserstellung (mit dem Ziel der vollautomatischen Ableitung von Werkstattzeichnungen aus 3D-Modellen) in solchen Schritten zu entwickeln, daß sich möglichst schnell Effekte auch beim Einsatz von 3D-fähigen CAD/CAM-Lösungen

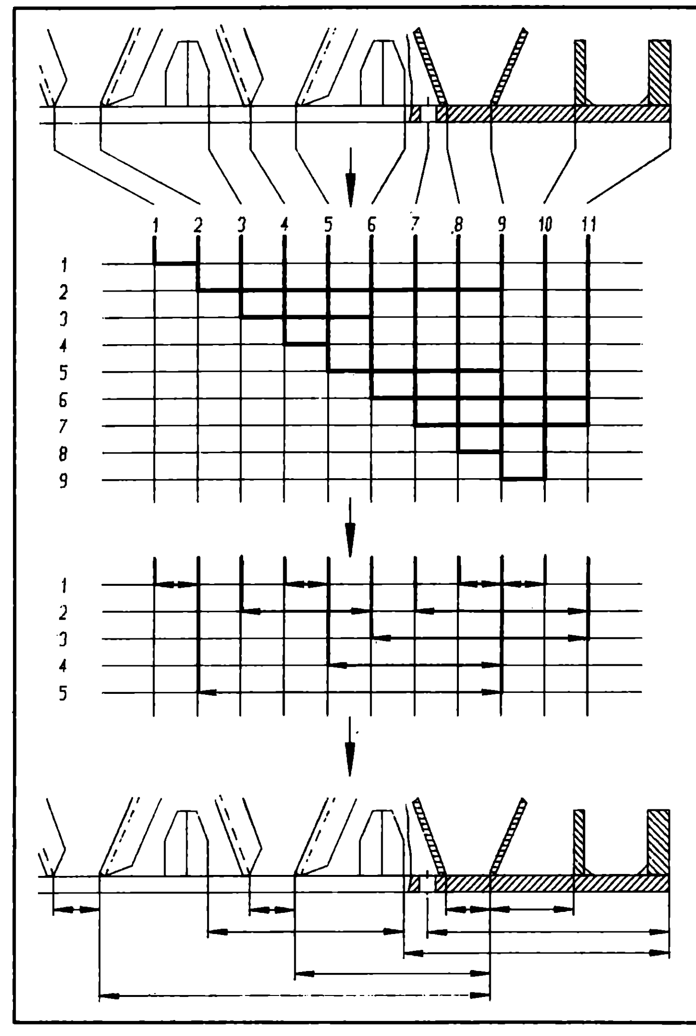
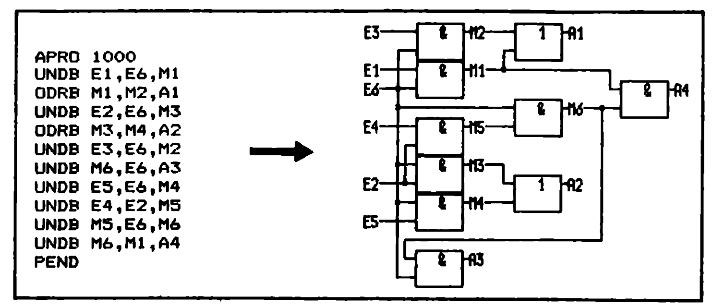


Bild 2. Illustration des automatischen Layouts ganzer Gruppen von parallelen Maßen

mit Modellrepräsentationen ergeben, die nicht allen o. g. Anforderungen genügen. Bei einem solchen Vorgehen wird der Systemnutzer zunächst von denjenigen Aktivitäten entlastet, die sein fachspezifisches Wissen am wenigsten fordern, nämlich von Layout-Problemen.

Einen Schritt auf diesem Weg stellen die am Zentralinstitut für Kybernetik und Informationsprozesse (ZKI) der AdW der DDR durchgeführten Arbeiten zum vollautomatischen Layout ganzer Gruppen von Maßen dar [9] und [13]. Es entstanden Verfahren und entsprechende Softwarebausteine, deren Input im wesentlichen eine Menge von Positionspaaren und deren Output vollständige parallele Bemaßungen sind. Bild 2 verdeutlicht Input und Output sowie die abstrakte (nicht geometrische) Modellbildung für den Prozeß der Reihenfolgeoptimierung schematisch. Eine wichtige Rolle spielen die im Bild 2 (mittlerer Bereich) waagerecht verlaufenden virtuellen Dimensionslinien (VDL), denen die einzelnen Maße durch entsprechende Algorithmen zur Reihenfolgeoptimierung zugeordnet werden. Der in [9] und [13] ausführlich vorgestellte

Bild 3. Einfaches Beispiel für die vollautomatische Erzeugung eines Logikplanes aus einer Steuerungsbeschreibung



(heuristische) Algorithmus A bestimmt sehr aufwandsarm eine in bezug auf die Anzahl von Linienkreuzungen günstige Reihenfolge. Da bei der Layout-Optimierung ganzer Bemaßungsblöcke außerdem das Ziel eines möglichst geringen Platzbedarfs besteht, wurde ein weiterer (exakter) Algorithmus entwickelt, der alle Bemaßungen mit polynomialem Aufwand so zusammendrückt, daß die Gruppe paralleler Maße insgesamt die minimal mögliche Anzahl von VDL belegt und damit in Richtung der Maßhilfslinien die geringste Ausdehnung hat, die prinzipiell erreicht werden kann. Da diese minimale Ausdehnung schon vor der Layout-Gestaltung sehr leicht bestimmt werden kann, ist mit diesem „Quetschalgorithmus“ (Algorithmus B in [9] und [13] eine wertvolle Grundlage für das automatische Auffinden von freien, nicht unnötig großen Flächenbereichen in der Werkstattzeichnung gegeben, in denen sich dann die jeweilige Gruppe paralleler Maße auch wirklich unterbringen läßt.

3. Automatisierte Erzeugung von grafischen Schemata

Wie bereits im Abschnitt 1. angedeutet, besteht bei netzartigen Schemata prinzipiell die Möglichkeit einer hochautomatisierten (bis hin zur vollautomatischen) Erzeugung aus solchen Beschreibungen, die lediglich die Struktur der Grafiken explizit oder implizit repräsentieren, aber nicht deren Layout spezifizieren. Im Rahmen einer mehrjährigen, im ZKI durchgeführten Entwicklung wurde zunächst für einen bestimmten Typ netzartiger Schemata untersucht, ob eine derartige Automatisierung praktikabel ist und welche Effekte sich dadurch erzielen lassen. Eine konkrete Anforderung in diese Richtung kam aus dem Automatisierungsanlagenbau. Es ging um das Erreichen eines möglichst hohen Automatisierungsgrades bei der Erzeugung von Logikplänen aus Quelltexten einer Fachsprache zur Beschreibung von Steuerungen, die in speicherprogrammierbaren Steuerungseinrichtungen (SPS) implementiert werden. Diese Aufgabe der Überführung einer funktionsorientierten Beschreibung in eine grafische Dokumentation wird durch Bild 3 anhand eines kleinen Beispiels angedeutet. Die hohe Komplexität dieser Aufgabe geht schon daraus hervor, daß bereits bestimmte Teilprobleme, wie die Trassierung von Verbindungslinien innerhalb der einzelnen Zwischenräume zwischen benachbarten Gruppen untereinanderliegender Symbole (das sogenannte Kanalrouting [14] und [15]), NP-vollständig sind [16] und [17]. Eine strenge Layout-Optimierung nach gewissen Kriterien erschien daher als nicht zweckmäßig, zumal die Optimierungskriterien selbst schwer formalisierbar sind. Auf der Basis heuristischer Algorithmen [15] [18] und [19] wurde ein Programmsystem (LOGIKPLAN) entwickelt, mit dessen Hilfe stets — korrekten Input vorausgesetzt — eine vollautomatische Erzeugung von Logikplänen möglich ist. Ein dabei erarbeitetes Verfahren zur Plazierung der Symbole ist außerdem in das System AUTODRAFT von *General Motors* eingegangen [20].

Bild 4 zeigt einen kleinen Ausschnitt aus einem mit LOGIKPLAN vollautomatisch generierten Logikplan.

Die Einsatzerfahrungen in der industriellen Projektierung lassen sich wie folgt zusammenfassen (s. auch [9]):

- Der Grad der Akzeptanz der vollautomatisch erzeugten Dokumentation ist so hoch, daß bereits Tausende von A3-Blättern mit Logikplänen in dieser Weise im Closed-shop-Betrieb produziert wurden.
- Dabei ergab sich eine Zeitreduktion gegenüber der manuellen Erstellung der Pläne auf der Basis von grafischen Vorlagen (Handskizzen) auf 1/60. Da der Aufwand für den Entwurf dieser Vorlagen bei der Anwendung von LOGIKPLAN ebenfalls entfällt, liegt der Produktivitätssteigerungsfaktor also sogar über 60 (s. Tafel 2).
- Neben dem hohen Rationalisierungseffekt ist bei der vollautomatischen Erstellung der Logikpläne von besonderer praktischer Bedeutung, daß genau das dokumentiert wird, was für die Implementation in SPS vorgesehen ist. Dadurch hat sich LOGIKPLAN als wertvolles Werkzeug für die Erkennung von Fehlern bei der Projektierung von Steuerungen erwiesen.

Bild 4. **Ausschnitt aus einem mit LOGIKPLAN vollautomatisch erzeugten Logikplan**

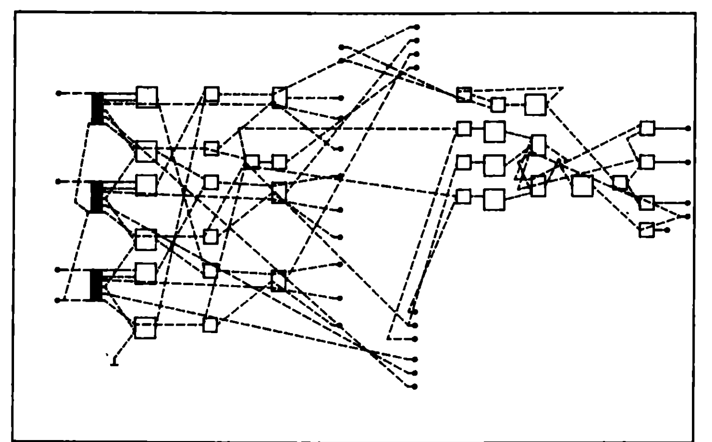


Bild 5. Interaktive Symbolplazierung und Netzeingabe

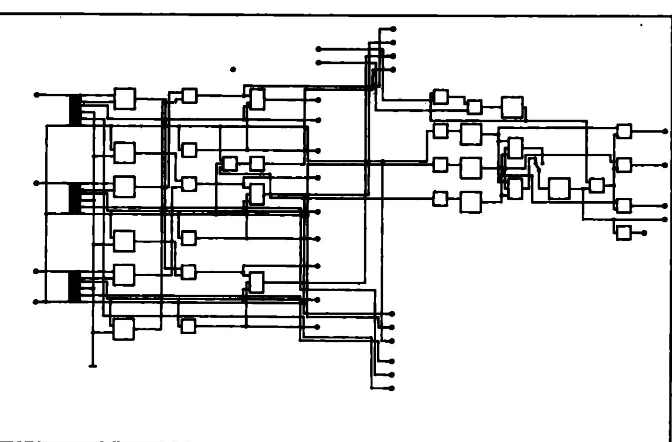


Bild 6. Automatisch durchgeführtes Routing der Verbindungslinien für die im Bild 6 dargestellte Spezifikation der Symbolplazierung und Verbindungsstruktur

- Da bei der Inbetriebnahme von Anlagen, i. allg. auch in SPS, viele Modifikationen vorzunehmen sind, ist die Erstellung von Revisionsdokumentationen bei konventioneller Zeichnungserstellung mit erheblichem Aufwand verbunden. Auch diesen kann LOGIKPLAN entsprechend reduzieren.
- Es scheint unmöglich zu sein, schematische Grafiken vollautomatisch zu erzeugen, ohne Wünsche offen zu lassen.

Der Nachweis der Realisierbarkeit einer vollautomatischen Erzeugung grafischer Schemata in einem konkreten Fall und die dabei erzielten praxisrelevanten Effekte waren die Grundlage für eine Ausweitung der Forschungs- und Entwicklungsarbeiten im ZKI auf dem Gebiet des Computer-Aided Schematics (CAS). Dabei geht es derzeit vor allem um eine Erweiterung der Anwendungsmöglichkeiten von Methoden und Hilfsmitteln zur weitgehend automatisierten Schematagenerierung. Die im letzten obigen Anstrich zusammengefaßten Erfahrungen führten zur Erarbeitung eines Konzeptes, dessen Umsetzung den Systemnutzer zukünftig in die Lage versetzen soll, den Grad der Einflußnahme auf die Gestaltung grafischer Schemata und damit auch den Grad der Automatisierung ihrer Erzeugung weitgehend selbst zu beeinflussen [2]. Die Einflußnahme soll auf verschiedenen Ebenen erfolgen können. Die unterste Ebene entspricht den Möglichkeiten konventioneller Systeme für die interaktive Erzeugung von Schemata (Beeinflussung der Grafik

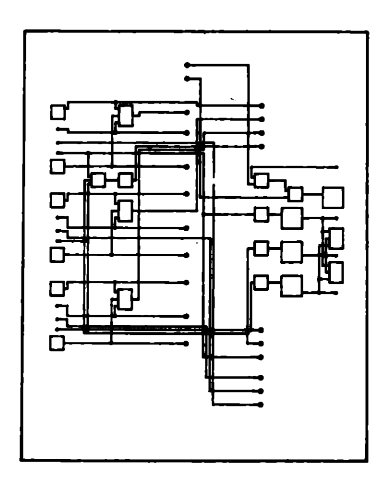


Bild 7.
Replazierung eines ganzen Blockes von Symbolen und Ausführung eines neuen automatischen Routing
Bild 7 bezieht sich auf einen Ausschnitt von Bild 5 bzw. 6

durch den Nutzer bis hinunter zu den grafischen Primitiven). Darüber liegt eine Ebene, auf der der Anwender lediglich die Plazierung von Symbolen beeinflußt und die Verbindungsstruktur (Netzlisten) eingibt. Das Routing der Verbindungslinien erfolgt automatisch. Das neue CAS-System hat diese Ausbaustufe bereits erreicht [21]. Die Bilder 5 und 6 vermitteln einen Eindruck davon.
Bild 5 zeigt die interaktiv eingegebene Plazierung von Symbolen (hier nur als leere Blöcke dargestellt) sowie die Struktur der Verbindungsnetze (durch gestrichelte Linien symbolisiert, die bei Festlegung der zu verbindenden Anschlußstellen von Symbolen automatisch eingeblendet werden). Im Bild 6 ist das Ergebnis nach Ausführung der vollautomatischen Trassierung und Generierung der Verbindungslinien dargestellt.
Besonders effektiv ist dieses System bei der Durchführung von Änderungen anwendbar: Häufig stellt sich erst nach weitgehender Fertigstellung eines netzartigen Schemas eine ungünstige Anordnung von Symbolen heraus. Bei der Nutzung konventioneller 2D-Systeme kann der Aufwand zur Änderung der Darstellung beträchtlich werden. Das automatische Routing gestattet dagegen eine äußerst rationelle Vorgehensweise. Lediglich die Plazierung der Symbole ist zu modifizieren, alles andere erfolgt automatisch, da natürlich die Informationen zur Verbindungsstruktur erhalten bleiben. Für einen Ausschnitt aus Bild 6 zeigt Bild 7 eine derartige Replazierung mit neu ausgeführtem automatischen Routing.

Literatur

[1] Cugini, U.; Ferri, G.; Mussio, P.; Protti, M.: Pattern-Directed Restoration and Vectorization of Digitized Engineering Drawings. Comput. & Graphics, 8 (1984) 4, pp. 337—350.
[2] Döring, S.; Heine, J.; Iwainsky, A.; May, M.; Meissner, A.; Vigerske, W.: CAS-System für Mikrorechnerautomatisierungsanlagen. Studie, ZKI, 1986.
[3] Spur, G.; Krause, F.-L.; Jansen, H.: Automatische Digitalisierung und Interpretation technischer Zeichnungen für CAD-Prozesse. Zeitschrift für wirtschaftliche Fertigung 81 (1986) 5, S. 235—241.
[4] Spur, G.; Krause, F.-L.; Jansen, H.; Timmermann, M.: Rekonstruktion von 3D-Modellen mit CASUS. Zeitschrift für wirtschaftliche Fertigung 82 (1987) 10, S. 569—574.
[5] Encarnação, J., u. a. (Hrsg.): CAD-Handbuch; Auswahl und Einführung von CAD-Systemen. Berlin/Heidelberg/New York: Springer-Verlag 1984.
[6] Chasen, S. H.: Formulation of System Cost-Efficienceness. In: Chasen, S. H.; Dow, J. W.: The Guide for the Evaluation and Implementation of CAD/CAM Systems. CAD/CAM Decisions, Atlanta/USA 1979.
[7] Eigner, M.; Maier, H.: Einführung und Anwendung von CAD-Systemen, Leitfaden für die Praxis. München/Wien: Carl Hanser Verlag 1982.
[8] Lipchin, L.: Managerial and Strategic Planning Aspects of CAD/CAM Technology Implementation. In: Proceedings of the 5th International Conference and Exhibition on Computers in Design Engineering. Butterworth 1982. pp. 23—43.
[9] Iwainsky, A.; Kaiser, D.; May, M.: Layout Problems in Computer-Aided Graphical Documentation. Proceedings of the V. Bilateral Workshop GDR-Italy with International Participation on Information in Manufacturing Automation. Dresden, July 22—24, 1987, pp. 286—305.
[10] Cooper, B. S.: CAD/CAM-Einsatz bei der Konstruktion und Fertigung von hydraulischen Schaltelementen (Teil 2). CAD/CAM 5 (1986) 1, S. 109—112.
[11] Eulenberger, K.: Rechnerunterstützte Konstruktion, Teil II: Grafisch-interaktives Detaillieren. Maschinenbautechnik, Berlin 33 (1984) 11, S. 597—601.
[12] Donner, J.; Wolf, E.: AUKOS — ein $2^1/_2$D-System zur rechnergestützten Herstellung von Maschinenbauzeichnungen. Neue Technik im Büro, Berlin 29 (1985) 6, S. 173—176.
[13] Iwainsky, A.; Kaiser, D.: Algorithmen zur Ermittlung günstiger Reihenfolgen von Maßlinien in Bemaßungsgruppen. ZKI-Report 870515-CAD3D-AUTZEI, 1987.
[14] Yoshimura, T.; Kuh, E. S.: Efficient Algorithmus for Channel Routing. IEEE Trans. on Comp.-Aided Design of Integrated Circuits and Syst. 1 (1982) 1, pp. 25—35.
[15] May, M.; Mennecke, P.: Layout of Schematic Drawings. Syst. Anal. Model. Simul. 1 (1984) 4, pp. 307—338.
[16] Garey, M. R.; Johnson, D. S.: Computers and Intractability — A Guide to the Theory of NP-Completeness. San Francisco/USA: Freeman 1979.
[17] Sahni, S.; Bhatt, A.: The Complexity of Design Automation Problems. In: Proc. 17th Design Automation Conf. 1980. pp. 402—411.
[18] May, M.; Iwainsky, A.; Mennecke, P.: Placement and routing for logic schematics. Computer-Aided Design 15 (1983) 3, pp. 115—122.
[19] May, M.: Computer-generated multi-row schematics. Computer-Aided Design 17 (1985) 1, pp. 25—29.
[20] Majewski, M. A.; Krull, F. N.; Fuhrman, Th. E.; Ainslie, P. J.: AUTODRAFT: Automatic Synthesis of Circuit Schematics. Proc. IEEE Int. Conf. on CAD 1986, pp. 435—438.
[21] May, M.; Döring, S.; Vigerske, W.: Routing in abstrakten Schemata. Tagungsbeiträge „Kombinatorik und Anwendungen", Lubmin, 6.—10. 4. 1987, S. 50—53.

msr 8953

V. Bauche, H. Friedrich, W. Heicking, H. Jarosch, D. Koch, U. E. Kriegel, I. Mohr, R. Rosenmüller

KI kompakt — EXPERT Common LISP

0. Einleitung

Das Eindringen des Computers in viele Bereiche, vor allem in fast alle Wissenschaftsdisziplinen, hat die Diskussion über die Entwicklung und den Einsatz geeigneter Software neu belebt. Insbesondere die Auswahl der Programmiersprache für die Formulierung einer Problemlösung erhitzt im Streitgespräch oft die Gemüter, hängt doch nicht zuletzt von ihr die Effektivität der Programmentwicklung und -abarbeitung ab.
Die Autoren sind weder in der Lage noch willens, eine Programmiersprache zu empfehlen, die für jeden Anwendungsfall optimal wäre. Alle Pläne zum Entwurf einer solchen Sprache scheiterten an der Vielschichtigkeit der zu modellierenden Probleme. Wenngleich von unterschiedlicher Qualität, so hat doch jede der existierenden Sprachen ihre Stärken und Schwächen, die bei der Auswahl stets ins Kalkül zu ziehen sind. Während beispielsweise die Sprache BASIC gut zur Einführung in den Umgang mit Computern geeignet ist, erweist sie sich als unzureichend für anspruchsvollere Programmieraufgaben. Programmiersprachen, wie PASCAL, COBOL oder FORTRAN, die sich gut zur Programmierung naturwissenschaftlicher und ökonomischer Berechnungen eignen, erlauben kein effektives Arbeiten mehr, wenn man sie zur Lösung von Problemen der künstlichen Intelligenz einsetzt.
Die künstliche Intelligenz (im weiteren KI) ist Ausdruck der bis vor einigen Jahren noch nicht vorauszusehenden Fähigkeiten der Rechentechnik, den Menschen auch in geistigen Arbeitsprozessen

zu unterstützen, wenn diese zwar menschliche Intelligenz erfordern, aber monotonen Charakter tragen, oder aber, wenn diese die Möglichkeiten des menschlichen Gedächtnisses überfordern. Wer die Softwareentwicklung der letzten Jahre verfolgt hat, dem wird nicht entgangen sein, daß Begriffe wie „Wissensverarbeitung", „Expertensysteme", „automatisches Schließen" usw. eine wachsende Rolle spielen.
Zur effektiven Realisierung solch komplizierter Prozesse auf dem Computer bedarf es komplexerer Ausdrucksmittel als sie die „klassischen" Sprachen zur Verfügung stellen. Es entstanden die sogenannten KI-Sprachen, von denen LISP und PROLOG die bekanntesten sind.
LISP gehört zu den ältesten Programmiersprachen (Definition 1958, erste Implementation 1960), ist aber mehr als 2 Jahrzehnte fast ausschließlich in Forschungszentren für erste experimentelle Entwicklungen in der künstlichen Intelligenz verwendet worden. Sowohl die Form (Syntax) als auch die benötigten Ressourcen haben den Einsatz von LISP für kommerzielle Zwecke erschwert. Wie heute deutlich zu erkennen, hat dieses Mauerblümchendasein von LISP nicht zu seiner Verkümmerung und schließlichem Tod geführt, sondern durch eine ungesteuerte aber intensive internationale Kooperation zu einer unvergleichlich hohen sprachlichen Reife. So entstanden Dialekte mit erweiterten Ausdrucksmitteln, einer leistungsfähigeren Arithmetik, einem Compiler, der die effektivere Ausführung der ursprünglich ausschließlich interpretativ abgearbeiteten LISP-Prozeduren ermöglichte. Damit konnten für LISP-Systeme Verarbeitsgeschwindigkeiten vergleichbar zu konventionellen Programmiersprachen erzielt werden, so daß auch zeitkritische numerische oder technische Berechnungen mit LISP möglich wurden.
Bei aller Unterschiedlichkeit gegenüber den herkömmlichen, auf numerische Aufgaben orientierten Sprachen muß LISP dennoch zu den klassischen Sprachen gezählt werden: Der gedachte Algorithmus muß wie eh und je ausprogrammiert werden. Daß dies häufig in einer kompakteren Form geschehen kann, liegt im wesentlichen an der Funktionalität und an der Angemessenheit von LISP für Aufgaben der Symbolmanipulation und bedeutet auch nicht, daß das Programmieren einfacher oder weniger zeitaufwendig ist. Jedoch kann man behaupten, daß der Gesamtzyklus der Programmentwicklung in LISP wesentlich kürzer ist als in den anderen klassischen Sprachen. Dies liegt daran, daß LISP nicht nur eine Sprache ist, sondern ein System, das alle Funktionen eines Betriebssystems (sprich: Programmentwicklungssystem) in sich vereinigt. Diese außergewöhnliche Mächtigkeit von LISP wird im wesentlichen durch die grundsätzlich interpretative Verarbeitung und durch die syntaktische Ununterscheidbarkeit von Daten und Programmen ermöglicht.
Bis zum Ende der 70er Jahre entstand eine große Vielfalt von Dialekten, die die Auswahl eines geeigneten Dialektes sehr erschwerte (zuweilen waren bei der Beseitigung von Mängeln neue entstanden) und manchen von der Sprache Abstand nehmen ließ. Standardisierungsbestrebungen Anfang der 80er Jahre führten 1984 zur Spezifikation eines neuen Dialektes Common LISP [1].
Common LISP ist in allen entwickelten kapitalistischen Ländern zu einem Quasi-Industriestandard geworden. In den USA, in Japan und in Westeuropa entstand eine Reihe sehr leistungsfähiger Implementationen, insbesondere auch für Personalcomputer und für LISP-Maschinen (Computer mit LISP-Spezialprozessoren). Die erste den Autoren bekannte Common LISP-Implementation der RGW-Länder wurde unter der Bezeichnung EXPERT Common LISP Anfang dieses Jahres im Bereich „Künstliche Intelligenz" am Zentralinstitut für Kybernetik und Informationsprozesse (ZKI) der AdW der DDR fertiggestellt. Common LISP ist ein sehr ausdrucksstarker, effizienter und eleganter Dialekt, der neben Konstrukten zur Symbolverarbeitung über eine ausgefeilte Arithmetik verfügt, die auch technische Anwendungen nahelegt. Anwendungen von LISP auf dem Gebiet der KI beinhalten u. a. die Entwicklung von CAD-Systemen, von Systemen zur Wissensrepräsentation und von Expertensystemen. Obwohl Common LISP eine gute Unterstützung bei der Erstellung von Prototyp-Systemen ermöglicht und für die Schaffung von KI-Software besser geeignet ist als die meisten anderen Programmiersprachen, gibt es Situationen, in denen komplexe KI-Methoden benötigt werden. Aus diesem Grunde werden in EXPERT Common LISP (im weiteren XCL) einige Werkzeuge implementiert, die besonders die Programmierung von Problemen der KI unterstützen:

— Werkzeuge für Vorwärts- und Rückwärtsinferenz mit PROLOG-Eigenschaften
— ein Produktionsregelsystem, vergleichbar mit OPS/5
— EXPERT Common TALK, ein Hilfsmittel zur objektorientierten Programmierung.

1. Implementation von EXPERT Common LISP

Eines der Ziele der XCL-Implementation war es, ein effizientes Common LISP-System auf Universalrechnern mit Wortbreiten von 32 bit als auch 16 bit zu schaffen. Deshalb wurden einige grundlegende Komponenten, wie Teile des Interpreters, des Parsers, einige Primitive zur Listenverarbeitung und Teile der Arithmetik, in der Programmiersprache C [2] implementiert. Die Behandlung von Pointern sowie einige Teile der Arithmetik hängen sehr stark von der internen Darstellung von Zahlen bzw. Pointern auf den entsprechenden Rechnertypen ab. Diese Teile wurden mit Hilfe von Konstanten bzw. Makros für den C-Preprozessor formuliert, der diese während der Compilation in den Quellkode einfügt. Abgesehen davon, daß Inline-Kode schneller abgearbeitet werden kann als ein Funktionsaufruf, bietet diese Technik die Möglichkeit, bei der Systemgenerierung lediglich durch die Spezifikation von etwa

Dipl.-Math. VOLKER BAUCHE (27) studierte von 1980 bis 1985 Mathematik an der TU Dresden. Seit 1985 wissenschaftlicher Mitarbeiter im Zentralinstitut für Kybernetik und Informationsprozesse (ZKI) der AdW der DDR, Bereich Künstliche Intelligenz.

Dr.-Ing. HORST FRIEDRICH (32) studierte nach einer Berufsausbildung (mit Abitur) als Facharbeiter für Datenverarbeitung von 1976 bis 1983 an der Sektion Informationsverarbeitung der TU Dresden, davon 3 Jahre Forschungsstudium, mit Abschluß Promotion A. Seit 1983 wissenschaftlicher Mitarbeiter am ZKI der AdW der DDR.

Dr. rer. nat. WINFRIED HEICKING (46) studierte von 1962 bis 1968 Physik an der Karl-Marx-Universität Leipzig. Von 1968 bis 1969 im VEB Fernmeldebereich Leipzig, 1969 bis 1970 Institut für Energetik, Leipzig. Seit 1970 wissenschaftlicher Mitarbeiter im ZKI der AdW der DDR. 1985 Promotion A an der AdW.

Dr.-Ing. HELMUT JAROSCH (41) studierte von 1966 bis 1971 Physik an der TU Dresden. Von 1972 bis 1988 wissenschaftlicher Mitarbeiter im Wissenschaftlichen Informationszentrum der AdW der DDR (Informationsrecherchesysteme und KI-Methoden in der Kommunikation mit Datenbanksystemen). 1979 Promotion A (Massendatenverarbeitung in Informationsrecherchesystemen). Seit 1988 wissenschaftlicher Mitarbeiter im ZKI der AdW der DDR. Lehrtätigkeit an der TH Ilmenau und der Humboldt-Universität zu Berlin auf dem Gebiet der theoretischen Grundlagen von Informationssystemen seit 1975.

Dr. sc. nat. DIETRICH KOCH (49) studierte von 1960 bis 1966 Physik. Von 1966 bis 1969 Aspirant an der Pädagogischen Hochschule Potsdam. 1969 Promotion A, 1982 Promotion B. Seit 1969 wissenschaftlicher Mitarbeiter im ZKI der AdW der DDR, Bereichsleiter „Künstliche Intelligenz".

Dr. rer. nat. ULRICH E. KRIEGEL (36) studierte von 1971 bis 1976 Physik. Von 1976 bis 1980 wissenschaftlicher Mitarbeiter im Institut für Hochenergiephysik der AdW, von 1980 bis 1982 im Institut für Geografie und Geoökologie der AdW. Seit 1972 wissenschaftlicher Mitarbeiter im ZKI der AdW der DDR. 1980 Promotion A.

Dipl.-Ing. INGO MOHR (32) studierte von 1977 bis 1982 an der Sektion Informationsverarbeitung der TU Dresden. Seit 1982 wissenschaftlicher Mitarbeiter im ZKI der AdW der DDR.

Dipl.-Math. RAINER ROSENMÜLLER (36) studierte von 1971 bis 1975 Mathematik an der Ernst-Moritz-Arndt-Universität Greifswald. Von 1975 bis 1978 Mitarbeiter im Institut für Nachrichtentechnik Berlin (Rechnergestützte Konstruktion von Leiterplatten). Seit 1979 wissenschaftlicher Mitarbeiter im ZKI der AdW der DDR.

20 Konstanten des Preprozessors die Eigenschaften des benutzten Rechnersystems einzustellen. Die Pointerstruktur in XCL wird hauptsächlich dadurch bestimmt, daß LISP über ein Laufzeit-Typsystem verfügt. Das heißt, daß es während der Laufzeit möglich ist, den Typ eines Objektes zu bestimmen und in Abhängigkeit von diesem Typ verschiedene Operationen durchzuführen. Typtests kommen sehr häufig vor und bestimmen damit wesentlich die Effizienz einer Implementation. Um schnelle Typtests zu unterstützen, wurde in XCL eine Architektur mit Typ-Tags emuliert — die Typ-Information eines Objektes ist im Pointer auf das Objekt enthalten. Die Pointer sind 32 bit lang und enthalten ein Adreßfeld von 24 bit, was einem Adreßraum von 16 Mbyte entspricht. Die restlichen 8 bit werden in ein 6 bit langes Typfeld und ein 2 bit langes cdr-Feld unterteilt. Letzteres dient der kompakten Abspeicherung von Listen. Die Wahl des aktuellen Typkodes ist willkürlich, mit Ausnahme des Kodes für kleine positive und negative ganze Zahlen, die für schnelle Arithmetik die Kodes 0 bzw. 63 belegen. Um eine größtmögliche Unabhängigkeit vom benutzten Rechnertyp zu sichern, wurden ein großer Teil von XCL selbst wie auch der Editor und der Compiler in XCL geschrieben. Nachdem der Compiler fertiggestellt war, wurde als nächster Schritt der Compiler compiliert. Letztlich werden dann die LISP-Funktionen compiliert, um ein besseres Laufzeitverhalten zu erreichen. Die Gesamtgröße des für das System zur Verfügung stehenden Hauptspeichers kann vom Nutzer in der Anlaufphase des Systems spezifiziert werden. Sie sollte jedoch nicht kleiner als 1 Mbyte sein. Um XCL auch auf 16-bit-Rechnern mit kleinem Hauptspeicher (etwa 1 Mbyte) zu unterstützen, wurde ein konventioneller Garbage-Collector mit Markierungs-, Lösch- und Verdichtungsphase implementiert. Für eine spätere Version von XCL ist ein inkrementeller Garbage-Collector für 32-bit-Rechner geplant. Die erste XCL-Implementation wurde für den K1840[1] mit dem Betriebssystem SVP generiert. Nach der Redefinition der oben erwähnten Preprozessor-Konstanten war es jedoch nicht schwierig, ein XCL-System für das P8000 zu generieren, ein 16-bit-Mikrorechnersystem des KEAW mit einem UNIX-artigen Betriebssystem. Es ist deshalb zu erwarten, daß auch die Erzeugung von XCL-Systemen auf anderen Rechnertypen relativ problemlos möglich ist.

2. XCL-Compiler

Um ein effizientes Laufzeitverhalten von XCL zu gewährleisten, wurde ein Compiler entwickelt, der LISP-Programme in semantisch äquivalente Maschinenprogramme übersetzt. Der XCL-Compiler besteht aus 2 Pässen. Der erste Paß transformiert LISP-Quellkode in eine maschinenunabhängige LISP-Assemblersprache. Die Elemente dieser Sprache sind reguläre LISP-Ausdrücke, die als Compiler- oder kurz C-Makros bezeichnet werden. Diese C-Makros werden im 2. Paß benutzt, um die Instruktionen für den jeweiligen Rechnertyp zu erzeugen. Damit der Compiler weitgehend unabhängig von der zugrundeliegenden Hardware ist, erfolgt die Optimierung des Kodes während des ersten Passes. Optimiert wird in 2 Schritten sowohl auf Quellkode- als auch auf C-Makro-Niveau. Zunächst wird der Quellkode lokal optimiert, um ungenutzte Berechnungen und Zuweisungen bzw. nicht zu erreichende Programmteile zu eliminieren. Danach erfolgt eine globale Optimierung, in der Restrekursionen und ein Teil der übrigen Rekursionen aufgelöst werden. Auf C-Makro-Niveau erfolgen ebenfalls zwei Optimierungsschritte, eine Peephole- und eine globale Optimierung. Der Compiler kann in 2 Modi benutzt werden. Der erste Modus dient dazu, einzelne LISP-Funktionen zu übersetzen und den erzeugten Maschinenkode sofort einzupflanzen (d. h. dynamisch zu verbinden). Im File-Modus übersetzt der Compiler alle Funktionen, die in einer Datei enthalten sind, und schreibt das Ergebnis — den Objektkode — auf eine Datei. Beim Einladen einer so erzeugten Objektdatei werden alle darin enthaltenen Funktionen automatisch eingepflanzt. Im folgenden werden

[1]) Natürlich hängt der 2. Paß des Compilers stark von der Architektur des Zielsystems ab und muß für jeden Rechnertyp neu implementiert werden.

Tafel 1. Vergleich der Rechenzeiten mit verschiedenen Programmiersprachen

Ackermann-Funktion
Zeiten in ms

x	y	PASCAL	C-Compiler	XCL-Compiler
3	4	40	40	20
3	5	160	170	80
3	6	730	770	330
3	7	2860	3370	1460

Tafel 2. Vergleich XCL-Compiler-/XCL-Interpreter/C-Version für die Funktion LAST für eine Liste mit 50000 Elementen

	C-Version	XCL-Interpreter	XCL-Compiler
CPU-Zeit in ms	130	23400	90
Platz in byte	301	116	129

einige vorläufige Laufzeittests wiedergegeben, die auf dem K1840 des ZKI ermittelt wurden. Die Möglichkeit, effektiven Kode für Belange zu erzeugen, die eine tiefe Schachtelung von Funktionsaufrufen erfordern, soll anhand der Ackermann-Funktion demonstriert werden, die in LISP wie folgt beschrieben wird:

```
(DEFUN ACM (X Y)
    (DECLARE TYPE FIXNUM X Y))
    (COND (( = X 0) (+ Y 1))
          (( = Y 0) (ACM (— X 1) 1))
          (T      (ACM (— X 1) (ACM X (— Y 1))))) .
```

Verglichen wird diese Funktion mit in PASCAL und C geschriebenen Varianten, die mit den Compilern des K1840 übersetzt wurden. Das Ergebnis des Tests ist in Tafel 1 angegeben.
In allen Fällen wird der vom XCL-Compiler erzeugte Kode doppelt so schnell abgearbeitet wie der von konventionellen Sprachcompilern erzeugte. Die Fähigkeit, schnellen Kode für symbolische Berechnungen zu generieren, wird am Beispiel der Funktion LAST

```
(DEFUN LAST(L &AUX (L1 L))
  (LOOP (SETQ L1 (CDR L1))
        (IF L1 (SETQ L L1)
               (RETURN L))))
```

demonstriert, die in Tafel 2 mit der im Kern von XCL enthaltenen, in C geschriebenen Version verglichen wird.
Verglichen mit der interpretierten Version erhält man eine Verbesserung von etwa 200 bei annähernd gleichem Platzbedarf. Eine Ursache für das gute Abschneiden des vom XCL-Compiler erzeugten Kodes liegt sicherlich darin, daß aufgrund der Parameterübergabe mittels Stack-Frames keine Rücksicht auf den für alle im Betriebssystem eingebundenen konventionellen Sprachen gültigen Call-Standard genommen werden mußte. Die Autoren sind sich klar darüber, daß Benchmarks mit kleinen Funktionen nur bedingt aussagefähig sind. Als ein komplexeres Beispiel wurde der Wangsche Algorithmus [3] zum Prüfen von Aussagen der Aussagen-Logik in LISP programmiert und compiliert, wobei jedoch nur eine provisorische Version des XCL-Optimizers eingesetzt werden konnte. Der compilierte LISP-Kode ist knapp 3mal langsamer als der durch einen PASCAL-Compiler erzeugte. Mit einer verbesserten Version des Optimizers dürfte sich das Verhältnis jedoch günstiger gestalten.
Die angeführten Beispiele zeigen, daß der XCL-Compiler Kode erzeugen kann, der in der Effizienz dem von Compilern konventioneller Sprachen erzeugten Kode vergleichbar ist. Genauere Aussagen zur Effizienz können zum jetzigen Zeitpunkt nicht gemacht werden. Im Zuge der XCL-Entwicklung werden deshalb weitere Benchmark-Tests mit publizierten Testfunktionen [4] durchgeführt, deren Ergebnisse einer späteren Publikation vorbehalten sind.

3. EXPERT Common TALK

Die auf dem Gebiet der KI zu lösenden Probleme können, wenn man an Expertensysteme, Wissensverarbeitung oder CAD-Systeme denkt, so komplex werden, daß selbst Common LISP (in seiner ursprünglichen Form) nicht ausreicht, um die Aufgaben effizient und in guter Qualität zu lösen. Deshalb wird XCL um eine objektorientierte Komponente erweitert, die die Implementation großer komplexer Systeme durch viele Programmierer effektiviert. Erreicht wird dies durch die hohe Modularisierung der Programme, wie sie objektorientierten Systemen eigen ist.

Im Gegensatz zur herkömmlichen Programmierung werden Daten (Objektvariablen, ähneln Record-Komponenten) und die sie manipulierenden Prozeduren (auch Methoden genannt) in einer untrennbaren Einheit, einem Objekt, zusammengefaßt. Die Implementationsdetails eines Objekts sind nach außen hin nicht sichtbar. Der einzige Weg, um mit einem Objekt zu kommunizieren und um seinen Zustand zu ändern, ist das Versenden einer Nachricht. Erhält ein Objekt eine Nachricht, so wählt es zur Beantwortung eine seiner Methoden aus und aktiviert diese. Methoden dienen sowohl zur Zustandsänderung als auch zur Ausführung von Berechnungen und der Abfrage von Informationen über den internen Zustand des Empfängers.

Das objektorientierte Konzept bewirkt eine starke Reduzierung der Abhängigkeiten zwischen den einzelnen Programmkomponenten, wodurch sich Programmiersicherheit, Schnelligkeit und andere Faktoren für die Produktivität der Softwareherstellung verbessern. Die Integrität eines Softwareprodukts bleibt z. B. bei Änderungen in der Datenrepräsentation i. allg. erhalten, da solche Änderungen nicht außerhalb eines Objekts sichtbar sind.

EXPERT Common TALK wird aus drei Teilen bestehen:

— der Basisebene
— einem Metasystem
— EXPERT Common FLAVORS.

Die Basisebene enthält im wesentlichen den Datentyp OBJECT und einen Satz von Funktionen zum Versenden von Nachrichten. Sie stellt die eigentliche objektorientierte Erweiterung von XCL dar, da der Rest von EXPERT Common TALK auf dieser Grundlage und natürlich unter Nutzung der Mittel des Common-LISP-Standards implementiert wird.

Die Objekte sind auf dieser Ebene weder in Klassen und Instanzen unterteilt noch in einem Vererbungsgraphen organisiert. Das zu realisieren, bleibt den speziellen Teilsystemen von EXPERT Common TALK vorbehalten. Ein Objekt ist hier nur eine Zusammenfassung aus einer Variablenumgebung, in der z. B. Methoden abgearbeitet werden, und einer Nachrichtenbehandlungsfunktion, die unter anderem die Vererbung und die Aktivierung von Methoden übernimmt. Ein Teilsystem von EXPERT Common TALK hat die Aufgabe, solche Nachrichtenbehandlungsfunktionen bereitzustellen und somit das allgemeine Verhalten seiner Objekte zu definieren.

EXPERT Common FLAVORS, eines der speziellen Teilsysteme von EXPERT Common TALK, ist ein objektorientiertes Programmiersystem, das den prozeduralen Aspekt der Objekte stark unterstützt und sich deshalb für solche Aufgaben wie Grafiksysteme, Fensterverwaltung oder Filemanagement besonders gut eignet. Zu diesem Zweck werden verschiedene Arten von Variablen und Methoden angeboten, die ganz bestimmte Geltungsbereiche haben und deren Verarbeitung so effektiv wie möglich gestaltet wird.

Die Objekte werden, wie in vielen ähnlichen Systemen, in Klassen (Platzhalter für Beschreibung des Aufbaus und des Verhaltens der Instanzen) und Instanzen („konkrete" Realisierungen anhand der Beschreibung in einer Klasse) unterteilt. Sie sind alle in einem Vererbungsgraphen angeordnet, dessen Terminalknoten die Instanzen sind. Entlang der Kanten dieses Graphen werden verschiedene Informationen zum Aufbau und Verhalten der Instanzen von allgemeinen zu spezielleren Klassen weitergeleitet (vererbt). Dies führt zu einer merklichen Reduzierung der Programmgröße und erhöht die Übersichtlichkeit, da oft benutzte Informationen nur einmal spezifiziert werden müssen.

Weitere, die Leistungsfähigkeit des Systems bestimmende Elemente, sind die Dämone und Filter. Dämone sind spezielle Methoden, die vor bzw. nach Abarbeitung einer Hauptmethode (i. allg. von einer allgemeineren Klasse geerbt) abgearbeitet werden, wobei sie die gleichen Argumente wie die Hauptmethode übergeben bekommen. Filter arbeiten ähnlich, jedoch werden sie auf die Argumente bzw. die Ergebniswerte der Methode angewendet, die somit leicht an geänderte Anforderungen angepaßt werden können. Mit Hilfe dieser beiden Arten von Spezialmethoden ist es möglich, geerbte allgemeine Methoden in einer Klasse so zu modifizieren, daß sie den spezielleren Anforderungen gerecht werden. So sind bei der Definition einer neuen Klasse oft nur die inkrementellen Änderungen gegenüber ihren Superklassen anzugeben, eine Neudefinition mit nur wenig geänderten Details ist unnötig. Das wirkt sich insbesondere auf die Kompaktheit, aber auch auf die Änderungsfreundlichkeit der Programme günstig aus.

Eine sehr wesentliche Eigenschaft des Systems wird darin bestehen, daß es vollständig in Common LISP integriert wird. So kann man (und muß sogar) alles das nutzen, was Common LISP bietet. Zum Beispiel sind die Methoden nichts anderes als LISP-Funktionen, die mit einer speziellen Umgebung, den Objektvariablen, arbeiten. Das bedeutet, daß das volle Spektrum der Parameterarten mit geforderten, wahlweisen, Schlüsselwort- und Restparametern von Common LISP zur Verfügung steht. Ebenso kann man die Variablen eines Objekts, wie von LISP gewohnt, nutzen, also einfach durch Angabe ihres Namens und nicht mittels irgendwelcher magischer Konstrukte, wie oft in auf LISP aufgesetzten Systemen.

EXPERT Common FLAVORS ist jedoch weniger für solche Aufgaben wie Wissenspräsentation oder Regelverarbeitung geeignet. Da es unmöglich ist, ein System zu schaffen, das den Anforderungen verschiedenartiger Problemklassen gleichermaßen gut gewachsen ist und dennoch für alle effizient arbeitet, wird man mit dem Metasystem die Möglichkeit haben, an bestimmte Aufgabenklassen speziell angepaßte, effizient arbeitende objektorientierte Teilsysteme zu implementieren. Die Metaebene bietet einen Rahmen für die Implementation solcher Teilsysteme und stellt einige Werkzeuge bereit, die diese Arbeit erleichtern. Der wechselseitigen Nutzung verschiedener dieser Systeme steht im Normalfall auch nichts im Wege, da alle auf den gleichen Grundlagen, der Basisebene und Common LISP, aufbauen.

Als weiteres spezielles Teilsystem von EXPERT Common TALK wird ein System entwickelt, das neben Methoden auch Regeln als Antwort auf eine Nachricht aktivieren kann. Damit sollen die Eigenschaften von LISP, regelbasierten Systemen und objektorientierten Programmiersprachen in einem Paket zur Verfügung stehen.

Literatur

[1] STEELE, G. H.: COMMON LISP — The Language. Digital Press 1984.
[2] KERNIGHAN, C. W.; RITCHIE, D. M.: The C Programming Language. Englewood Cliffs: Prentice-Hall 1978.
[3] WANG, H.: Towards mechanical mathematics. IBM Journal of Research and Development 4 (1960), pp. 2—22.
[4] GABRIEL, R. P.: Performance and Evaluation of LISP Systems. Cambridge, Mass./London: The MIT Press 1985.

msr 9003

C. Clauß, U. Donath, G. Kurth, P. Schwarz, P. Trappe, H. Wittwer

KOSIM — ein Multi-Level-Simulator für den Schaltkreisentwurf[1]

0. Einleitung

Simulationsprogramme sind wichtige Hilfsmittel beim rechnergestützten Entwurf von Schaltkreisen. Dabei lassen sich zwei Anwendungsgebiete unterscheiden:

a) Beim Top-down-Entwurf werden nacheinander Simulationen in den verschiedenen Entwurfsetappen (Systementwurf, Logikentwurf, Entwurf der Transistorschaltung) ausgeführt. Dabei simuliert man vor allem die neu entworfenen Baublöcke und ihr Zusammenwirken mit bereits vorhandenen.
b) Zur Verifikation des Schaltkreisentwurfs wird, ausgehend vom Layout, eine Rückerkennung des elektrischen oder logischen Netzwerkes vorgenommen und der gesamte Schaltkreis simuliert [1] und [2].

KOSIM unterstützt beide Aufgaben, ist aber vor allem für eine Multi-Level-Simulation bestimmt. Darunter versteht man die Simulation von Systemen, deren Teilsysteme auf unterschiedlichen Beschreibungsniveaus modelliert und zusammen simuliert werden [3] bis [8].

Anwendungsgebiete der Multi-Level-Simulation mit KOSIM sind:

— im Top-down-Entwurf entstehende Schaltkreisbeschreibungen, deren Teile auf unterschiedlichen Niveaus vorliegen (z. B. Blockschaltbild, Gatterplan, Transistorschaltung)
— analog-digital arbeitende Schaltkreise
— sehr große Schaltkreise, bei denen sich Teile durch stark vereinfachte Modelle beschreiben lassen (PLA, ROM, RAM usw.)
— gemeinsame Simulation des Schaltkreises und seiner „Umgebung", also der Leiterplatte oder des Gerätes
Dabei wird man die Umgebung meist auf einem hohen, relativ abstrakten Niveau beschreiben.

[1] Überarbeitete Fassung eines Vortrages auf der European Conference on Circuit Theory and Design (ECCTD), Paris 1987

Dipl.-Math. CHRISTOPH CLAUSS (38) studierte von 1970 bis 1974 Mathematik. Von 1974 bis 1978 Industrietätigkeit. Seit 1978 wissenschaftlicher Mitarbeiter im Zentralinstitut für Kybernetik und Informationsprozesse (ZKI) der AdW der DDR, Institutsteil Dresden.
Dr.-Ing. ULRICH DONATH (40) studierte von 1968 bis 1972 Theoretische Elektrotechnik an der TH Ilmenau. Von 1972 bis 1976 wissenschaftlicher Assistent an der Ingenieurhochschule Dresden, Sektion Informationsverarbeitung. 1977 Promotion A. Seit 1976 wissenschaftlicher Mitarbeiter im ZKI der AdW der DDR, Institutsteil Dresden.
Dipl.-Ing. GUNTER KURTH (47) studierte von 1961 bis 1967 Regelungstechnik an der TU Dresden. Nach Tätigkeit im VEB Robotron Meßelektronik Dresden seit 1984 wissenschaftlicher Mitarbeiter im ZKI der AdW der DDR, Institutsteil Dresden.
Dr.-Ing. PETER SCHWARZ (47) studierte von 1959 bis 1964 Elektrotechnik/Hochfrequenztechnik an der TU Dresden. Von 1964 bis 1967 dort Aspirantur. 1968 Promotion A. 1967 bis 1981 wissenschaftlicher Mitarbeiter im VEB Robotron-Meßelektronik (vorher WTZ Elektrotechnische Meßtechnik). Seit 1982 im ZKI der AdW der DDR, Institutsteil Dresden, wissenschaftlicher Abteilungsleiter.
Dr.-Ing. PETER TRAPPE (48) studierte von 1959 bis 1965 Elektrotechnik/Hochfrequenztechnik an der TU Dresden. 1975 Promotion A. Von 1965 bis 1968 wissenschaftlicher Assistent an der TH Karl-Marx-Stadt. Anschließend bis 1970 wissenschaftlicher Mitarbeiter im VEB Halbleiterwerk Frankfurt/O. Von 1971 bis 1982 desgl. im VEB Robotron-Meßelektronik Dresden. Seit 1983 wissenschaftlicher Mitarbeiter im ZKI der AdW der DDR, Institutsteil Dresden.
Dipl.-Ing. HARALD WITTWER (30) studierte von 1981 bis 1986 Informationstechnik an der TH Karl-Marx-Stadt. Seither Entwicklungsingenieur im Forschungszentrum Mikroelektronik Dresden (ZMD).

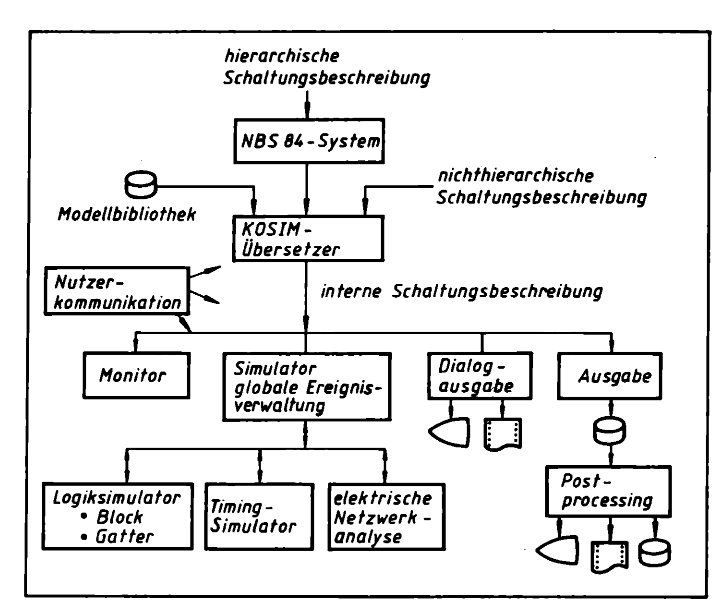


Bild 1. Programmstruktur

1. Überblick über das Programm

KOSIM (Bild 1) ist vor allem für die Simulation digitaler MOS-Schaltkreise gedacht. Es enthält Algorithmen für

— funktionelle Simulation (Logikblockniveau)
— Logikgattersimulation
— Timing-Simulation
— Netzwerkanalyse.

Das Modell der realen Schaltung ist ein Netzwerk, bestehend aus Elementen und Knoten (Verbindungen zwischen den Elementen). Die hierarchische Beschreibung der Schaltungsstruktur erfolgt mit der Netzbeschreibungssprache NBS84 [9]. Vor der eigentlichen Simulation wird das hierarchische Netzwerk in ein nichthierarchisches Netzwerk umgewandelt.
Sämtliche Algorithmen arbeiten ereignisorientiert. Ein „Ereignis" ist eine Signaländerung auf einem Knoten, gelegentlich auch die Änderung eines internen Elementezustands. Es werden nur die aktiven Schaltungsteile simuliert, in denen gerade Signaländerungen ablaufen. Alle Signaländerungen werden in einem ganzzahligen Zeitraster beschrieben. Dieser kleinste Zeitschritt (minimum resolvable time, *MRT* [7]) beträgt gewöhnlich 1 ns oder weniger. Die Verzögerungszeiten der Logikelemente sind ganze Vielfache dieses Zeitschrittes. Bei der elektrischen Simulation werden (gemäß der automatischen Schrittweitensteuerung) auch Rechnungen an Zeiten zwischen den Rasterzeitpunkten ausgeführt. Die Kopplung zwischen den elektrischen Teilschaltungen untereinander und mit den Logikelementen, also die Auswertung von Signaländerungen in benachbarten Schaltungsteilen, erfolgt nur an den Rasterzeitpunkten. Auf diese Weise werden die Simulationsalgorithmen synchronisiert.
Alle während der Simulation entstehenden Ereignisse, also künftige Signaländerungen, werden in einer Ereigniswarteschlange QUEUE gespeichert. Für einen schnellen Zugriff ist sie in zweifacher Weise organisiert:

— Ereignisse in der nahen Zukunft werden in einer matrixähnlichen Weise registriert („Zeile" = Zeitpunkt $i \cdot MRT$, „Spalte" = Nummer des Knotens, an dem zur Zeit $i \cdot MRT$ voraussichtlich eine Signaländerung stattfinden wird). Für diese schwachbesetzte Matrix wird eine Listenstruktur angelegt.

— Ereignisse in der ferneren Zukunft werden in einer nach der Zeit sortierten Liste gemerkt und in die erste Liste umgespeichert, wenn dies erforderlich wird.

Zusammen mit dem ganzzahligen Zeitraster (die Simulationszeit T wird durch eine INTEGER*4-Variable beschrieben) ist damit eine sehr effektive Ereignisverwaltung möglich.
Auf den verschiedenen Modellierungsebenen werden unterschiedliche Signaltypen benutzt: elektrische Spannungen, logische Bit-Signale, komplexe Datentypen (Wort-Signale) wie INTEGER, REAL oder CHARACTER auf dem Logikblockniveau. Zwischen den entsprechenden Schaltungsteilen müssen vom Anwender Konverter eingefügt werden, die eine Signaltransformation ausführen (DAC, ADC, Bit-Wort-Konverter, Wort-Bit-Konverter).

2. Elektriksimulation

Es können zwei Simulationsalgorithmen verwendet werden:

— die sehr genauen, aber langsamen Verfahren der nichtlinearen Netzwerkanalyse [10] bis [12]
— die wesentlich schnellere, aber ungenauere Timing-Simulation [1] [7] [11] [13] bis [15].

2.1. Netzwerkanalyse

Zur Formulierung der Netzwerkgleichungen wird die modifizierte Knotenspannungsanalyse verwendet. Als Integrationsverfahren für die steifen Algebro-Differentialgleichungssysteme hat sich die Trapezregel als ausreichend genau erwiesen. Da meist nur über ein kurzes Zeitintervall (MRT) integriert wird, lohnt die aus den GEAR-Algorithmen bekannte Ordnungssteuerung nicht. Die elektrische Schaltung, die mit der Netzwerkanalyse simuliert werden soll, wird vom Anwender in Blöcke zerlegt, die möglichst gut entkoppelt sein sollen. Diese Blöcke werden einzeln über das Intervall MRT integriert, der Einfluß der Nachbarblöcke wird durch linearisierte Ersatzelemente nachgebildet. Dabei werden Blöcke nicht simuliert, in deren Innerem und an deren Randknoten in einem Zeitintervall keine Signaländerungen vorliegen (Latenzprinzip). Mit diesem Verfahren der „blockorientierten Netzwerkanalyse" [30] konnten Schaltungen mit 25000 Transistoren berechnet werden.
Es stehen Transistormodelle verschiedener Genauigkeit (auch SPICE-ähnliche Modelle) zur Verfügung, die an unterschiedliche nSGT- und CSGT-Technologien angepaßt sind. Parasitäre Kapazitäten werden ebenfalls durch nichtlineare Modelle beschrieben. Neue Bauelementemodelle lassen sich als „Nutzerprozeduren" einfach in den Simulator einfügen (Linken erforderlich), da Schnittstellen dafür vorbereitet sind.

2.2. Timing-Simulation

Timing-Simulation ist ein iteratives Verfahren der Netzwerkanalyse, bei dem durch

— Entkopplung der Knoten und Lösen der skalaren Differentialgleichungen der einzelnen Knoten
— wenige Iterationen (u. U. nur eine) über alle Knotengleichungen an einem Zeitpunkt
— Ausnutzen der Latenz einzelner Knoten oder Gruppen von Knoten

viel Rechenzeit gespart wird, natürlich auf Kosten der Lösungsgenauigkeit. Numerische Probleme, die zu einer größeren Anzahl von Iterationen führen, gibt es bei eng gekoppelten Knoten (durch „transmission gates" oder „floating-C" verbunden). In diesen Fällen werden besser kleine Gleichungssysteme anstelle einzelner Gleichungen gelöst.
Als Integrationsverfahren für die einzelnen Differentialgleichungen kann wahlweise die Euler-Rückwärtsformel oder eine „Exponentialangleichung" verwendet werden. Die Spannungsänderung an einem Knoten in einem Zeitschritt h beträgt bei der Euler-Formel

$$\Delta U = \frac{I}{C/h + G}, \qquad G = -\frac{\mathrm{d}I}{\mathrm{d}U},$$

wobei I der Strom durch die an Erde liegende Knotenkapazität C ist. Bei der Exponentialangleichung werden in einem Zeitintervall alle Zweige an einem Knoten durch ihre Norton-Äquivalente (Stromquelle, Leitwert, Kapazität) ersetzt. Für die Berechnung dieser Größen werden die jeweils neuesten iterierten Werte der Knotenspannungen verwendet (Gauss-Seidel-Iteration). Die Spannungsänderung in einem Zeitschritt h ist

$$\Delta U = R_{\mathrm{equ}} \; I_{\mathrm{equ}} \cdot [1 - \exp(-h \cdot G_{\mathrm{equ}}/C_{\mathrm{equ}})] .$$

Es wird also für jeweils ein Zeitintervall das lineare Ersatznetzwerk in der Umgebung eines Knotens exakt berechnet, ein wesentlicher Unterschied zur Euler-Rückwärtsformel. Wegen der geringeren Genauigkeit der Timing-Simulation werden nur einfache Transistormodelle und lineare Kapazitäten verwendet.

3. Logikgattersimulation

KOSIM benutzt auf dem Logikniveau Algorithmen der ereignisorientierten dynamischen Logiksimulation [16]. Die Elemente führen eine Verknüpfung der Eingangssignale durch und stellen die zugehörigen Ausgangssignale bereit. Für die Signaländerung am Elementeausgang ist jedem Element wenigstens eine Verzögerungszeit $DELAY$ zugeordnet. Die Knoten übernehmen die Signale von den Elementeausgängen und leiten sie den Eingangsklemmen der nachfolgenden Elemente zu. Signalverknüpfungen führen Knoten nur in wenigen Fällen durch. Dies sind die Operationen für Wired-OR oder Wired-AND und Busoperationen für Elemente mit Tristate-Ausgang. Grundsätzlich sind die Datenstrukturen aber so angelegt, daß jedem Knoten (in Analogie zu den Elementen) eine Signalverarbeitungsprozedur zugeordnet werden kann.
Der verwendete Algorithmus für die ereignisorientierte dynamische Logiksimulation lautet [17]:

1. Suchen in der Warteschlange QUEUE nach dem nächsten Zeiteintrag T
2. Aktualisierung der Knotensignale
   Für alle in der QUEUE zur Zeit T eingetragenen Knoten k:
   a) Übertragung des zur Zeit T gültigen Signals S vom Elementenausgang zum Knoten k
      Bei Wired-Knoten erfolgt eine Verknüpfung der Ausgangssignale aller speisenden Elemente (Vorgängerelemente) unter Berücksichtigung der Signalstärke.
   b) Falls keine Signaländerung vorliegt, dann Streichen des Knotens k aus der QUEUE zur Zeit T
3. Aktivierung der Folgeelemente
   Für alle in der QUEUE zur Zeit T eingetragenen Knoten k:
   a) Ermittlung der Folgeelemente n des Knotens k
   b) Für alle Folgeelemente n:
      Berechnung aller Ausgangssignale S_i und Zeiten $T_i = T + DELAY_i$ ihres Auftretens an den Ausgangsklemmen, Eintrag aller Folgeknoten i zur Zeit T_i in die QUEUE
4. Übergang zu 1.

Für die Logiksimulation reicht bei modernen Schaltungstechniken eine zweiwertige Signaldarstellung (0, 1) nicht aus. Gewählt wird eine Darstellung in Form eines Tupels:

Signal = (Zustand, Wert),
Wert ∈ {0, 1, X}
Zustand ∈ {LOW_IMPEDANCE R = 0,
RESISTIVE R = 1 ... 126,
HIGH_IMPEDANCE,
UNKNOWN}

„Wert" ist die logische Information auf der Verbindung. „Zustand" charakterisiert das elektrische Widerstandsniveau bzw. die Signalstärke am Elementeausgang. Die Trennung von „Wert" und „Zustand" ermöglicht eine dreiwertige Logik für Gatter und Flip-Flops. Bei den Verknüpfungsoperationen dieser Elemente wird „Zustand" ignoriert. Für Busse, die mehrere Tristate-Elemente als Quellen haben, und für Knoten mit bidirektionalen Elementen ist „Zustand" entscheidend: LOW-IMPEDANCE-Signale überschreiben HIGH_IMPEDANCE- und RESISTIVE-Signale, RESISTIVE-

Signale mit einem kleinen R-Wert überschreiben HIGH_IMPEDANCE-Signale und RESISTIVE-Signale mit einem größeren R-Wert.
Die Verknüpfungsoperationen der Elemente werden in Verhaltensprozeduren definiert. Für folgende Elemente stehen eingebaute Modelle bereit:
OR, NOR, AND, NAND, XOR, EQV, BUFFER, INVERTER, Latch (RS, D), Flip-Flop (RS, D, JK), Master-Slave-Flip-Flop (RS, D, JK), uni- und bidirektionales Transfergate, Verzögerungsleitung, Lasttransistor, Tristate-Buffer und -Inverter, ROM, RAM, PLA, Signalquellen (einschließlich eines „Patterngenerators" mit automatischem Vergleich von Soll- und Istwerten).

Für den Patterngenerator wurde in Anlehnung an die PATSY-Datenstrukturen eine Patternbeschreibung LSPAT für die dynamische Logiksimulation eingeführt. Es ist auch möglich, das Verhalten von Elementen durch Boolesche Gleichungen zu beschreiben. Weiterhin sind alle Elemente des Standardzellensystems U1500 des VEB Zentrum für Mikroelektronik Dresden (ZMD) im Modellvorrat von KOSIM enthalten.
Die Programmierung und Einführung von Verhaltensprozeduren neuer Elemente ist leicht möglich und wird durch Zugriffsroutinen auf die simulatorinternen Daten unterstützt.

4. Logikblocksimulation

Die Logikblocksimulation ist eine Erweiterung der Gattersimulation auf komplexe Elemente („Blöcke") wie ALU, ROM, RAM, PLA, DECODER usw. [17]. Auf dem Blockniveau sind die Verbindungen zwischen den Elementen Leitungsbündel, die als Einheit behandelt werden. Sie haben eine Datenbreite von n bit. Daher läßt sich die Logikblocksimulation auch als Logikwortsimulation auffassen, im Gegensatz zur Logikbitsimulation auf dem Gatterniveau.
Für den Simulationsalgorithmus (siehe oben) ist diese Erweiterung ohne Belang, da dieser im wesentlichen nur Signaldaten vom Elementeausgang zum Knoten und vom Knoten zum nächsten Elementeeingang „transportiert" (abgesehen von Wired-Verbindungen). Die Verarbeitung der Signaldaten erfolgt in den Verhaltensprozeduren der Elemente.
Die Signale werden ebenfalls als Tupel definiert:

Signal = (Zustand, Wert),
Wert $\in \{0, 1\}^n$, Bitkette der Länge n
Zustand $\in$ {ND = LOW_IMPEDANCE, DEFINED
ZD = HIGH_IMPEDANCE, DEFINED
NU = LOW_IMPEDANCE, UNDEFINED
ZU = HIGH_IMPEDANCE, UNDEFINED}

„Wert" ist eine Bitkette der Länge n, die vielfältig interpretiert werden kann. So wird „Wert" als INTEGER- oder REAL-Zahl direkt als Operand in logischen oder arithmetischen Ausdrücken der Elementeprozeduren verwendet. „Zustand" charakterisiert das elektrische Widerstandsniveau der Signalquelle und wird zur Realisierung von Busoperationen benötigt. Außerdem enthält „Zustand" die Aussage, ob ein definiertes Signal vorliegt. „Zustand" bezieht sich auf das Signal als Einheit, kann also für einzelne Bit nicht unterschiedlich sein. Auf dem Blockniveau sind u. a. folgende Elemente im Modellvorrat enthalten:

ROM, RAM, PLA, Multiplexer, Bit-Wort- und Wort-Bit-Konverter; NOT, NOR, AND, NAND, XOR, EQV, ADD, SUB, SHIFT, ROTATE; Signalquellen.

Mit diesen Elementen lassen sich bereits viele Blöcke in Schaltkreisen modellieren. Andere Blöcke müssen vom Nutzer als neue Funktionsblöcke definiert werden, deren Verhalten durch Prozeduren beschrieben wird.
In KOSIM gibt es drei Möglichkeiten zum Definieren neuer Verhaltensprozeduren für Logikblöcke: zwei prozedurale Beschreibungsformen und eine nichtprozedurale Hardwarebeschreibungssprache.

4.1. FORTRAN-Subroutinen

Die Signalverarbeitung im Inneren des Funktionsblockes wird durch Befehle der Programmiersprache FORTRAN 77 modelliert. Neben der eigentlichen Berechnung des Blockverhaltens sind die Übernahme von Daten aus dem Simulator in die Nutzerprozedur und die Übergabe von neuen Werten an den Simulator zu organisieren. Im Interesse einer klaren Schnittstellenfestlegung verfügt jede FORTRAN-Subroutine über folgenden Aufbau [17] und [18]:

P1. Vereinbarung der internen Variablen
P2. Lesen der Eingangssignale des Funktionsblockes und seiner Parameterwerte (Zählgrößen, Zustandsvariable, Tabellen usw.)
P3. Berechnung von Ausgangssignalen und neuen Parametereinstellungen
P4. Schreiben der Parameterwerte und Ausgangssignale und Planen der Signalwechsel an den Elementausgängen entsprechend der vorgegebenen Verzögerungszeiten.

Der direkte Zugriff auf die simulatorinternen Daten ist dem Nutzer nicht gestattet. Das Lesen der Eingangssignale, Schreiben der Ausgangssignale sowie das Lesen und Schreiben von Parameterwerten erfolgt über Zugriffsroutinen, die im Simulator bereitstehen.
Die Formulierung der Verhaltensprozeduren in FORTRAN bietet größte Flexibilität, erfordert aber vom Nutzer gute Programmierkenntnisse. An Stelle von FORTRAN könnten (vor allem für den „Kern" P3 der Prozedur) auch andere Programmiersprachen, wie PASCAL oder C, verwendet werden.

4.2. Prozedurale Blockbeschreibung mit BBS

Um eine einfache, hardwarenahe Beschreibung eines Logikblockes ausführen zu können, wurde eine „Blockbeschreibungssprache" BBS eingeführt. BBS ist eine prozedurale, FORTRAN-ähnliche Sprache, in die einige Elemente der Hardwarebeschreibungssprache DDL aufgenommen wurden. Es gibt keinen Bezug auf simulatorinterne Daten.
Im Kopf der Blockbeschreibung werden alle Eingangs- und Ausgangssignale sowie internen Signale und Register mit ihren Namen und ihrer Datenbreite (z. B. REG (15:0)) aufgelistet. Weiterhin wird die Verzögerungszeit für die Signalwechsel an den Blockausgängen definiert. Der Körper der Blockbeschreibung besteht aus einer Folge von Signalfluß- und Steueranweisungen. Es werden die Signalflußanweisungen

— connection: signal_name = expression
— transfer: register_name ← expression
— proc_call: CALL procedure (argument_list)

unterschieden.
Jede Connection-Anweisung (=) wird unmittelbar, verzögerungslos ausgeführt, während jede Transfer-Anweisung (←) an das Taktsignal des Registers gebunden ist.
Als Steueranweisungen stehen die Anweisungen

— IF-THEN-ELSE
— DECODE-CASE
— BOOLEAN
— EXIT

zur Verfügung.
Eine Blockbeschreibung in BBS muß in eine FORTRAN-Subroutine übersetzt werden. Dabei werden die Anschlußbedingungen an den Simulator hergestellt. Zur Zeit existiert für die BBS noch kein Compiler, so daß die „Übersetzung" in eine FORTRAN-Nutzerprozedur durch den Nutzer (mit Hilfe einer Transformationsvorschrift) selbst erfolgen muß. Trotzdem lohnt sich die Notation des Blockverhaltens in BBS, da sie eine übersichtliche und kompakte Beschreibung gestattet.
Eine genaue Beschreibung von BBS ist in [18] angegeben.

4.3. DDL-Beschreibung

Das Innere eines Blockes kann auch mit der Hardwarebeschreibungssprache DDL beschrieben werden. Durch einen DDL-Translator [19] erfolgt eine Umformung in Boolesche Gleichungen und Re-

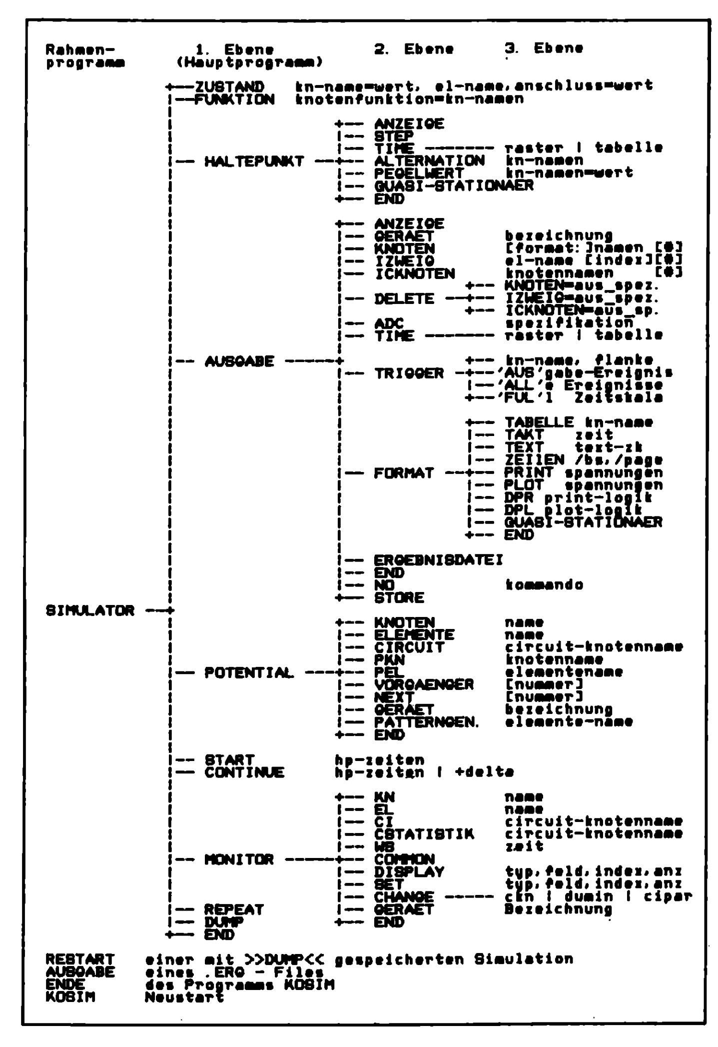


Bild 2. Kommandoübersicht

gister-Transfer-Anweisungen. Das Translatorergebnis wird in eine FORTRAN-Subroutine gemäß Abschn. 4.1. umgesetzt. Dazu wurde ein eigenes Programmsystem HBD geschaffen [20] und [29], mit dem aus einer NBS-Beschreibung, in der auch ein oder mehrere Blöcke durch DDL beschrieben werden, die FORTRAN-Subroutinen automatisch generiert und in das KOSIM-Programmsystem eingebunden werden. Durch die Kopplung von HBD und KOSIM ist eine Multi-Level-Simulation unter Einbeziehung des Register-Transfer-Niveaus möglich. Darüber hinaus kann das HBD-System auch eine reine DDL-Simulation ohne Multi-Level-Umgebung und daher auch ohne KOSIM vornehmen.
Aus den drei Beschreibungsformen kann der Anwender diejenige auswählen, die seinen Vorstellungen und dem zu simulierenden System am besten angepaßt ist. In jedem Fall entsteht eine neue FORTRAN-Subroutine, die übersetzt und in den Simulator eingebunden wird. Ähnliche erfolgreiche Experimente zur Kopplung anderer Beschreibungssprachen [21] und Simulationssysteme [22] und [23] mit KOSIM haben gezeigt, daß das Konzept von Nutzerprozeduren sehr tragfähig ist.

5. Nutzerkommunikation

KOSIM ist ein dialogorientiertes Simulationsprogramm, das jedoch auch in der Stapelverarbeitung lauffähig ist. Mit Hilfe seiner hierarchisch aufgebauten Steuerkommandos kann der Ablauf der Simulation und die Ausgabe der Simulationsergebnisse in vielfältiger Weise beeinflußt werden. Bild 2 zeigt einen Überblick über die wichtigsten Kommandos.
Auf allen Kommandoebenen stehen HELP-Informationen zur Verfügung.

Ein Beispiel für eine einfache Kommandofolge ist:

```
SIM
AUSGABE KNOTEN INPUT(*), OUT(1), OUT(2), CLOCK
        END
ZUSTAND SIG1=0, S12=1
START 10000.
```

Kommandos können zusammengefaßt und in einer Kommandodatei abgelegt werden. Besonders wichtig sind die Kommandos zur Gestaltung der Ausgabe und zur Haltepunktsteuerung (ähnlich wie bei Logikanalysatoren). Neben der Ausgabe ausgewählter Größen in Tabellenform (Bitmuster, Zahlenkolonnen, Print-Plot, ...) auf Bildschirm und/oder in ein File können sämtliche Ergebnisse in ein Ergebnisfile zur späteren Auswertung (post processing) abgelegt werden.
Für eine nachträgliche Ergebnisauswertung sind zwei Möglichkeiten vorgesehen:

- Das Ergebnisfile wird noch einmal vom Simulator ausgewertet, es stehen daher fast sämtliche Kommandos des Simulators (z. B. für die Gestaltung der Dialogausgabe oder der Haltepunktsteuerung) zur Verfügung.
- Der Nutzer schreibt sich ein eigenes Programm zur Auswertung des Ergebnisfiles, dessen Datenstrukturen bekannt sind. Dadurch lassen sich individuelle, der jeweiligen Aufgabe besonders gut angepaßte Formen der Ausgabesteuerung und der Ergebnisdarstellung realisieren.

Für Nutzer, die eigene Verhaltensprozeduren schreiben, ist der Monitor ein wichtiges Hilfsmittel beim Test. Er gestattet die Anzeige und temporäre Änderung aller internen Daten. Ein Pfadverfolger erleichtert während der Simulation die Fehlersuche im Netzwerk. Mittels der Kommandos DUMP und RESTART ist es möglich, eine Simulation bis zu einem bestimmten Zeitpunkt auszuführen, einen Dump durchzuführen und die Simulation später fortzusetzen.

6. Anwendungen des Programms

KOSIM konnte erfolgreich beim Entwurf verschiedener Schaltkreise eingesetzt werden. Dazu gehören Vertreter aller üblichen Klassen von Schaltkreisen:

- Prozessoren, die im freien Entwurf (full-custom design) entstehen [24] und [25]
- Standardzellen-Schaltkreise:
  KOSIM wird im Standardzellensystem U1500/U1600 des VEB Zentrum für Mikroelektronik Dresden (ZMD) für die Logikgattersimulation eingesetzt.
- Gate-Array-Schaltkreise [26]
- Speicherschaltkreise (statische und dynamische RAM).

Dabei wird KOSIM sowohl als „Ein-Ebenen-Simulator" (z. B. für die dynamische Logiksimulation auf Gatterniveau im U1500-System oder als Timing-Simulator beim Speicherentwurf) als auch als Multi-Level-Simulator (z. B. mit Logikblock- und Logikgattersimulation in [24]) benutzt. Die Analog-Digital-Simulation (auch als Hybrid- oder Mixed-mode-Simulation bezeichnet) wurde bisher relativ selten angewendet (da die meisten mit KOSIM simulierten Schaltkreise rein digital arbeiten). Ein Modellierungsbeispiel ist in [27] angegeben. In der Zukunft wird sich das aber wohl ändern. In [28] wird für den Beginn der 90er Jahre international bei Kundenwunschschaltkreisen (ASIC) ein Anteil von 50% für digital-analog arbeitende Schaltkreise prognostiziert und daraus die Notwendigkeit von Mixed-mode-Simulationen abgeleitet.
KOSIM ist auf einem 32-bit-Rechner K 1840 (Robotron) implementiert. Es können z. Z. Schaltungen mit maximal 32000 Knoten und 32000 Elementen simuliert werden. Da ein „Element" auf den oberen Modellierungsebenen z. B. ein ROM oder ein Funktionsblock ALU sein kann, der eine Schaltung mit vielen tausend Transistoren nachbildet, ist mit KOSIM die Simulation von LSI- und VLSI-Schaltkreisen möglich.

Den ursprünglichen Modellvorrat von KOSIM haben vor allem Dr. J. Haufe (ZKI) durch die Einbeziehung Boolescher Gleichungen, Dipl.-Ing. A. Komor (ZMD) durch Standardzellenmodelle und Dr. M. Krauss durch Transistormodelle für die Timing-Simulation erweitert. In die Programmentwicklung sind viele Anregungen von Anwendern eingeflossen. Dafür danken die Autoren den Schaltkreisentwerfern im ZMD, im ZKI, im Kombinat Robotron, an der TU Dresden und in weiteren Betrieben.

Literatur

[1] Hecker, W.; Rössler, F.; Möschwitzer, A.: LSINET — Ein neues Logik- und Timingsimulationsprogramm für LSI- und VLSI-Schaltkreise. Nachrichtentechnik-Elektronik, Berlin 34 (1984) 6, S. 214 bis 218.

[2] Sasaki, T., u. a.: MIXS: A mixed level simulator for large digital systems. Proc. 17th Design Autom. Conf. 1980.

[3] Borrione, D.; Humbert, M.; Le Faou, C.: Hierarchical mixed-mode simulation mechanisms in the CASCADE project. In: F. Anceau; E. J. Aas (Hrsg.): VLSI '83. Amsterdam/New York. North-Holland Publ. Comp. 1983.

[4] Daseking, H. W.; Gardner, R. I.; Weil, P. B.: VISTA: a VLSI CAD system. IEEE Trans. CAD-1 (1982) 1, pp. 36—52.

[5] Doshi, M. H.; Sullivan, R. B.; Schuler, D. M.: THEMIS logic simulator — a mix mode, multi-level, hierarchical, interactive digital circuit simulator. Proc. 21. Design Autom. Conf. 1984, pp. 24—31.

[6] Reynaert, Ph.; DeMan, H., u. a.: DIANA: a mixed-mode simulator with a hardware description language for hierarchical design of VLSI. Proc. ICCC 1980, pp. 356—360.

[7] Newton, A. R.: Techniques for the simulation of large-scale integrated circuits. IEEE Trans. CAS-26 (1979) 9, pp. 741—749.

[8] Arnout, G.; DeMan, H.: The use of threshold functions and boolean-controlled network elements for macromodelling of LSI circuits. IEEE J. Solid-State Circuits SC-13 (1978) 3, pp. 326—332.

[9] Issel, W., u. a.: NBS-84: A structural description language for VLSI design. Proc. ECCTD '85, Prag 1985, pp. 62—65.

[10] Chua, L. O.; Lin, P.-M.: Computer-aided Analysis of Electronic Circuits. Englewood Cliffs: Prentice-Hall 1975.

[11] Antognetti, P.; Pederson, D. O.; DeMan, H.: Computer Design Aids for VLSI Circuits. Alphen: Sijthoff & Nordhoff 1981.

[12] Clauss, C.: Netzwerkanalyse als Bestandteil der Multi-Level-Simulation. 32. Int. Wiss. Koll. TH Ilmenau 1987, Sektion B2, S. 171—174.

[13] Saleh, R. A.; Kleckner, J. E.; Newton, A. R.: Iterated timing analysis. Proc. ICCAD 1983, pp. 139—140.

[14] Newton, A. R.; Sangiovanni-Vincentelli, A. L.: Relaxation-based electrical simulation. IEEE Trans. CAD-3 (1984) 4, pp. 308—330.

[15] Schwarz, P.: Timing-Simulation beim Schaltkreisentwurf. 32. Intern. Wiss. Koll. TH Ilmenau 1987, Sektion B2, S. 245—248.

[16] Szygenda, S. A.; Thompson, E. W.: Digital logic simulation in a time-based, table-driven environment. Computer (1975) 3, pp. 24—36.

[17] Donath, U.; Schwarz, P.; Trappe, P.: Dynamische Logiksimulation auf Bit- und Wort-Niveau. 1. Tagung Schaltkreisentwurf, TU Dresden 1986, S. 218—228.

[18] Donath, U.; Schwarz, P.: Funktionelle Simulation beim Schaltkreisentwurf. Postervortrag, 32. Int. Wiss. Koll. TH Ilmenau 1987, Sektion B 2.2. (und ZKI-Forschungsbericht).

[19] Breuer, M. A. (Hrsg.): Digital System Design Automation: Languages, Simulation & Data Bases. Computer Science Press 1975.

[20] Garte, D.; Haufe, J.; Ruelke, S.: Nutzerhandbuch des HBD-Systems. ZKI 1987.

[21] Müller, J.; Lattermann, J.: PASHDL — Eine Möglichkeit zur funktionellen Hardware-Beschreibung. Wiss. Zeitschrift der TU Dresden (1989) 1.

[22] Crenze, U.: Einige Aspekte der Implementierung und Anwendung der Hardwarebeschreibungssprache ISPS. 16. ZKI-Arbeitstagung, Dresden 1988, S. 75—79.

[23] Fenzke, P.: Realisierung eines Simulatorgenerators auf der Basis der Hardwarebeschreibungssprache ISPS. 16. ZKI-Arbeitstagung, Dresden 1988, S. 80—84.

[24] Klein, W.: Entwurf eines digitalen 16-Bit-Signalprozessors. 16. ZKI-Arbeitstagung, Dresden 1988, S. 63—66.

[25] Oberst, E.; Wächter, F.: Erarbeitung des funktionellen Verständnisses eines komplexen Schaltkreises durch Modellierung und Simulation. 16. ZKI-Arbeitstagung, Dresden 1988, S. 67—70.

[26] Engelmann, F. E.; Schäffer, D.; Sühnel, C.: Gate-Array-Schaltkreise für den Satellitenrundfunk. 16. ZKI-Arbeitstagung, Dresden 1988, S. 58—62.

[27] Kurth, G.; Trappe, P.: Ein Programmsystem zur multi-level-Simulation integrierter Schaltkreise. 32. Intern. Wiss. Koll. TH Ilmenau 1987, Sektion B2, S. 207—210.

[28] Goering, R.: A full range of solutions emerge to handle mixed-mode simulation. Computer Design (1988) 3, pp. 57—65.

[29] Garte, D.; Haufe, J.; Ruelke, S.: HBD — Ein Programmsystem zur Simulation von Struktur und Funktion komplexer digitaler Systeme. 2. Int. Seminar Informationstechnik, Hochschule für Verkehrswesen, Dresden 1989, S. 73—77.

[30] Clauss, C.; Schwarz, P.: Ein Verfahren zur blockorientierten Analyse nichtlinearer Netzwerke. 3. Tagung Schaltkreisentwurf, Dresden 1989, S. 153—157.

msr 9006

In eigener Sache

Wir suchen einen Fachredakteur für unsere Zeitschrift, der in oder bei Berlin über Wohnraum verfügt. Er sollte ein abgeschlossenes Hochschulstudium in der Fachrichtung Technische Kybernetik/Automatisierungstechnik oder in einem angrenzenden Gebiet (z. B. Technische Informatik oder Elektrotechnik/Elektronik) und gute Kenntnisse der deutschen Sprache haben.

Seine Aufgaben umfassen:

— rédaktionelle und fachliche Bearbeitung von Manuskripten
— Besuch und Auswertung von Fachtagungen, -messen und -ausstellungen
— fachbezogene journalistische Tätigkeit.

Interessenten wenden sich bitte an:

VEB Verlag Technik
Oranienburger Str. 13/14
Berlin 1020

Redaktion „messen · steuern · regeln"
Tel.: 2 87 03 62
2 87 03 74

Informationen zu neuen internationalen Vorschriften der Elektro- und Automatisierungstechnik; Teil 2[1]

IEC 185/186 (Ausgabe 1987)

„Stromwandler/Spannungswandler“

Die überarbeiteten IEC-Publikationen gelten für Wandler für elektrische Meßgeräte und Schutzeinrichtungen, die im Frequenzbereich von 15 bis 100 Hz arbeiten und die mit separaten Wicklungen ausgestattet sind. Die IEC 186 ist unterteilt in
— generelle Anforderungen an alle Spannungswandler
— Forderungen an einphasige Meßwandler
— zusätzliche Anforderungen an einphasige Schutzwandler
— zusätzliche Anforderungen an kapazitive Spannungswandler.

Der Aufbau der IEC 185 ist ähnlich dem der IEC 186. Unterschiede sind gegeben bei den dielektrischen Prüfungen, z. B. durch Schaltvorgänge hervorgerufene Kurzzeit-Überspannungen für Wandler über 300 kV, sowie der Einführung einer neuen Prüfmethode mit niedriger Prüfspannung bei längerer Prüfdauer und speziellen Teilentladungsmessungen.

IEC 146-4 (Ausgabe September 1986)

„Halbleiter-Stromrichter; Teil 4: Verfahren zur Festlegung des Betriebsverhaltens und der Prüfanforderungen von unterbrechungsfreien Stromversorgungssystemen (USV)“

In dieser IEC-Publikation sind Festlegungen zu Anlagen für die unterbrechungsfreie Stromversorgung mit Halbleiter-Stromrichtern und mit Energiespeichern (Akkumulatorenbatterien) in Gleichstromzwischenkreisen erfaßt. Die Vorschrift enthält
— Begriffe und deren Erläuterungen
— Aussagen zur Anlagenkonfiguration und Systembeschreibung
— Definitionen der normalen und besonderen Betriebsbedingungen
— Prüfverfahren
— Kennwerte für die Funkentstörung
und ist auch für andere USV-Anlagen mit anderen technischen Lösungen anwendbar.

IEC 605-4 (Ausgabe 1986)

„Prüfung der Zuverlässigkeit von Geräten; Teil 4: Schätzwerte und Vertrauensgrenzen“

Im Teil 4 der IEC 605 werden rechnerische und grafische Verfahren empfohlen, mit deren Hilfe nach Zuverlässigkeitsprüfungen Schätzwerte und Vertrauensgrenzen von Zuverlässigkeitskenngrößen für Geräte ermittelt werden können. Dies betrifft hauptsächlich die Kenngrößen Ausfallrate, mittlerer Ausfallabstand (MTBF) und Erfolgsquotient. Die empfohlenen Verfahren sind nicht nur bei besonders durchgeführten Zuverlässigkeitsprüfungen anwendbar, sondern auch während bzw. im Anschluß an Betriebsversuche und im normalen Betrieb. Für den häufig angenommenen Fall der konstanten Ausfallrate wird unterschieden in durch Zeitablauf bzw. Ausfallanzahl beendete Prüfung. Die Vertrauensgrenzen werden hierbei mit Hilfe der Chi-Quadrat-Verteilung ermittelt. Tabellen und grafische Verfahren sollen die Auswertung erleichtern und für den Fall einer über die Zeit nicht konstanten Ausfallrate werden in der Richtlinie die Weibull- und Normalverteilung dargestellt. Weiterhin ist in der IEC-Publikation die Berechnung des Schätzwertes und der Vertrauensbereich des Erfolgsquotienten enthalten. Der Erfolgsquotient ist als Wahrscheinlichkeit, mit der ein Versuch erfolgreich sein wird, definiert.

IEC 50 (Ausgaben 1986—87)

„Internationales Elektrotechnisches Wörterbuch (IEV)“

In den Jahren 1986 und 1987 wurden weitere Kapitel des IEV neu erarbeitet bzw. ergänzt. Dazu gehörten:
Abschnitt 603 Netzplanung und Netzführung elektrischer Energie mit Unterteilung der 149 Begriffe in die Bereiche Netzplanung, Berechnung, Stabilität, Netzführung, Netzzuverlässigkeit und wirtschaftliche Optimierung.

ISO 2382 (Ausgaben 1985—87)

„Datenverarbeitung; Begriffe“

Erschienen sind in den Jahren 1985 bis 1987 die Teile
1. Allgemeine Grundbegriffe
5. Darstellung von Daten
12. Periphere Geräte
19. Analog- und Hybridrechentechnik.

Weitere Teile sind z. Z. in Vorbereitung. Dazu gehören:
17. Textverarbeitung
24. Robotertechnik
25. Lokale Netze.

IEC 884-1 (Ausgabe 1987)

„Stecker und Steckdosen für den Hausgebrauch und ähnliche Zwekke; Teil 1: Allgemeine Anforderungen“

Die in der 1. Ausgabe der IEC-Publikation aufgeführten Anforderungen (technische Daten) und durchzuführenden Prüfungen gelten für alle verwendeten Stecker und Steckdosen für Wechselstrom bis 32 A und 440 V für normale Anforderungen. Diese Publikation soll möglichst kurzfristig als (West-)Europäische Norm (EN) eingeführt werden.

DIN/VDE 0847 Teil 4 (Ausgabe Januar 1987)

„Meßverfahren zur Beurteilung der elektromagnetischen Verträglichkeit; Störfestigkeit gegen gestrahlte Störgrößen“

Es werden im Teil 4 der BRD-Norm Meßverfahren zur Beurteilung der Störfestigkeit von elektrotechnischen Einrichtungen gegenüber gestrahlten Störgrößen angegeben. Um eine notwendige elektromagnetische Verträglichkeit zu erreichen, ist es erforderlich, daß die in Anlagen eingesetzten Geräte einen normierten Störpegel ohne Fehlfunktion ertragen können. Dazu werden im Teil 4 von DIN/VDE 0847 Meßverfahren zur Ermittlung der Störfestigkeit gegen magnetische, elektrische und elektromagnetische Felder in verschiedenen Frequenzbereichen angegeben und erläutert.

DIN/VDE 3546 (Ausgaben März bis August 1987)

„Konstruktive Gestaltung von Prozeßleitwarten“

Diese BRD-Richtlinie gilt für die konstruktive Gestaltung von Prozeßleitwarten für verfahrenstechnische Anlagen.

[1] Teil 1 erschien im H. 4/1989, S. 172—173.

Blatt:
— 01 gibt einen Überblick zur Wartengestaltung und über Grundbegriffe der Wartentechnik
— 02 enthält Empfehlungen für die Anwendungen bautechnischer Abmaße
— 03 behandelt die Anordnung der für die Prozeßführung benötigten Geräte in Tafeln und Pulten und gibt Empfehlungen zur Konstruktion von Instrumententafeln und Fließbildern
— 04 enthält Regeln für die Auslegung der Einbauten in Wartennebenräumen
— 05 erläutert die Anordnung von Bildschirmen in Warten.

DIN/IEC 38 (Ausgabe Mai 1987)

„IEC-Normspannungen"

Die Einführung dieser IEC-Norm in der BRD ergab nachfolgende wichtige Veränderungen:
— Im Niederspannungsbereich sind für DS-Netze nur noch der Nennbereich 230/400 V (statt bisher 220/380 V) zugelassen. Energieversorgungsunternehmen sind bereit, diesen NS-Wert einzuführen, da es zu keiner Beeinträchtigung der Lebensdauer und Sicherheit der in Anlagen eingesetzten Betriebsmittel kommen soll.
— Zulässige Mittelspannungsbereiche 3, 6, 10 und 20 kV (30 kV entfällt) für Betriebsmittel mit einer oberen Spannungstoleranz bis 20%.
— Zulässige Hoch- und Höchstspannungswerte 110, 220, 420 kV (statt bisher 380 kV) mit einer oberen Spannungstoleranz bis 10%, obere Höchstspannungsgrenze ist 1200 kV.

VDI 3641 (Entwurf Oktober 1986)

„Mobile Datenübertragungssysteme für den innerbetrieblichen Transport"

Im innerbetrieblichen Transport ist die Kommunikation zwischen der Leitstelle und dem eingesetzten Fahrzeug über mobile Erfassungsgeräte unerläßlich. Dies wird heute durch neue Technologien und durch den schnellen Datenaustausch zwischen den stationären und mobilen Rechnern sowie den Einsatz von Bildschirmeinheiten möglich. Der VDI-Entwurf enthält daher notwendige Informationen über EDV-unterstützende Kommunikationstechniken zur Lager- und Transportrationalisierung in Unterteilung zu den Systemen
— der leitungsgebundenen Datenübertragung mittels induktivem Fernsprechnetz
— der Infrarot-Datenübertragung
— mittels Datenfunk.

Auswertung

Nach der ersten Auswertung der in den Jahren 1986/87 von internationalen IEC-Arbeitsgruppen durchgeführten Standardisierungsarbeiten für Elektro- und Automatisierungsanlagen kann festgestellt werden, daß diese sich besonders aktiv auf nachfolgende Bereiche konzentrierten:
— Sicherheit von MSR-Geräten (Tagung der IEC-SC 66E im Oktober 1986 in Dubrovnik/Jugoslawien);
— Aufbau, Zuordnung, Dokumentation, Programmierung und Qualitätsanforderungen für Mikroprozessorsysteme (Tagung der IEC-SC 47 im September 1986 in Stockholm);
— Gerätebestückung, Wartenkomplettierung und Sicherheitssysteme beim Reaktoreinsatz (Tagung der IEC im Juli 1987 in Pittsburgh/USA).

Dieser Trend wird sich international in den nächsten Jahren verstärkt fortsetzen.

Der Beitritt des ASMW der DDR im Januar 1988 als Mitglied in die Internationale Organisation für Standardisierung (ISO) sollte eine stärkere Mitwirkung bei der Gestaltung bzw. Auswertung von neu herausgegebenen internationalen Vorschriften bewirken. Wichtige exportbestimmende Festlegungen müssen auch bei der Fertigung von DDR-Erzeugnissen schneller umgesetzt werden.

Literatur

[1] Kloust, H.: Informationen zu neuen internationalen Vorschriften der Elektro- und Automatisierungstechnik. msr, Berlin 29 (1986) 12, S. 558—561.
[2] Kloust, H.: Informationen zu neuen internationalen Vorschriften der Elektro- und Automatisierungstechnik. msr Berlin 31 (1988) 1, S. 39 bis 40.
[3] DIN-Mitteilungen: Neues aus dem Normenwerk. 66 (1987) 1 bis 12. Berlin/Offenbach: VDE-Verlag GmbH.
[4] etz: Normenanzeiger. etz 108 (1987) 1 bis 20.
[5] Warner, A.: Jahrbuch zum VDE-Vorschriftenwerk 1987. Neues über VDE-Bestimmungen und VDE-Leitlinien. Berlin/Offenbach: VDE-Verlag GmbH 1987.
[6] Mitteilungen des ASMW: Internationale Standards. Berlin 14 (1987) 1 bis 24. Verlag für Standardisierung.

msr 8891 Ing. Ing. H. Kloust, leitender wissenschaftlicher Mitarbeiter und Koordinierungsbeauftragter für Standardisierungsaufgaben der Automatisierungstechnik im ZFT des VEB Kombinat Automatisierungsanlagenbau Berlin

Es geht noch einfacher!

Bezugnehmend auf [1] soll im folgenden eine noch einfachere Approximation des Studentschen t-Faktors angegeben werden.

In der Praxis wird der t-Faktor für sehr kleine Freiheitsgrade k in der Regel nicht benötigt. Beschränkt man sich daher auf $k > 4$ und setzt eine Potenzfunktion zur Näherung an, wobei der Endwert $t(\infty)$ additiv hinzugefügt wird, so ergeben sich mit Hilfe eines Regressionsprogramms die folgenden Gleichungen:

Vertrauensniveau 95%

$$t_{95\%} = 4{,}3/k^{1,2} + 1{,}96 \tag{1}$$

Vertrauensniveau 99%

$$t_{99\%} = 10{,}73/k^{1,25} + 2{,}58\,. \tag{2}$$

Für $k = 1$ und $k = 2$ sind diese Gleichungen unbrauchbar. Der Fehler liegt aber bei (1) bereits ab $k = 4$ unter 1%, bei (2) ab $k = 5$. Damit stehen zwei einfache Näherungsbeziehungen für die Bereitstellung des t-Faktors in Computerprogrammen zur Verfügung.

Wenn der t-Faktor auch für $k = 1$ und $k = 2$ benötigt wird, kann man z. B. für das 95%-Vertrauensniveau folgende zwei Programmzeilen (BASIC) vorsehen:

```
100 IF K>2 THEN T=4.3/K∧1.2+1.96 : GOTO 120
110 IF K=2 THEN T=4.3 : ELSE T=12.71
120 ...
```

Der maximale Fehler ergibt sich hierbei für $k = 3$ zu ungefähr 2%; ansonsten liegt er durchweg bei bzw. unter 1%.

Für die Berechnung mit dem Taschenrechner benutzt man besser die Gleichungen:

$$t_{95\%} = 2{,}6/k + 1{,}96\,, \tag{3}$$

$$t_{99\%} = 6/k + 2{,}58\,. \tag{4}$$

Ihr Fehler liegt zwar erst ab $k = 8$ unter 1%, dafür sind sie aber so einfach, daß man sie zur Not auch im Kopf ausrechnen kann.

[1] Netzband, D.: Eine einfache Approximation des Studentschen t-Faktors mit Rechnern. msr, Berlin 31 (1988) 9, S. 416—417.

msr 9030 Prof. Dr.-Ing. habil. P. Vielhauer, Forstweg 33, Freital, 8210

R. Wieland

LISP — eine Einführung für Automatisierungstechniker 5

Ein Überschreiten der vereinbarten Feldgröße (dim) eines Feldes führt zu einer Fehlermeldung.

(AREF X 12) ⇒ Fehler
Die Länge eines Feldes läßt sich feststellen mit

(ARRAY-LENGTH vector) ⇒ Feldlänge
(ARRAY-LENGTH x) ⇒ 10

Eine Besonderheit des Zugriffs auf Feldelemente besteht in der Arbeit mit Zeigern. Dazu muß ein Zeiger bei einer Felddefinition vereinbart worden sein. Der aktuelle Stand des Zeigers kann mittels:

(FILL-POINTER vector) ⇒ Zeigerposition
(FILL-POINTER y) ⇒ 0

abgefragt werden. Mit SETF kann der Zeiger an eine entsprechende Stelle des Feldes positioniert werden.

(SETF (FILL-POINTER vector) position)

Über die Funktionen:

(VECTOR-POP vector) ⇒ vector/zeiger/
(VECTOR-PUSH z vector) ⇒ zeiger

kann auf Feldelemente zugegriffen werden. Dabei gibt VECTOR-POP das Element zurück, auf das der Zeiger wies. VECTOR-PUSH bringt z an die vom Zeiger positionierte Stelle und gibt den Stand des Zeigers zurück. Anschließend wird der Zeiger inkrementiert.

Beispiele:

```
(VECTOR-PUSH 3 y)    ⇒ 0
(VECTOR-PUSH 5 y)    ⇒ 1
(FILL-POINTER y)     ⇒ 2 ; aktueller Zeiger
(AREF y 0)           ⇒ 3 ; Zugriff mittels Feldindex
(AREF y 1)           ⇒ 5
(VECTOR-POP y)       ⇒ 3 ; dekrementiert den Zeiger
(FILL-POINTER y)     ⇒ 1
```

Die Arbeit mit Zeigern wird in ähnlicher Form auch in der Programmiersprache C verwendet, bietet also dem „C-Programmierer" nichts wesentlich Neues. Bemerkt werden soll dazu noch, daß auch Listen in Feldern gespeichert werden können. Die Adressierung über die Adressen bleibt dem Nutzer aber verborgen, er wird somit von der Organisation der Adreßrechnung entlastet. Wurde also z. B. im Feld x die Liste (a b c) an der Stelle 0 gespeichert, so erhält der Nutzer auf eine Abfrage die Liste und nicht die Adresse der Liste.

```
(SETF (AREF x 0) '(a b c))
(AREF x 0)           ⇒ (a b c)
```

2.4.2. ASSOC-Listen

Ausgangspunkt der nachfolgenden Einführung der ASSOC-Listen sei folgende Aufgabe: In einer verfahrenstechnischen Anlage sei eine Liste aus Meßgrößen und den ihnen zugeordneten Werten in folgender Form gespeichert:

((Meßgröße1 Wert1)(Meßgröße2 Wert2)...(Meßgrößek Wertk)).

Diese Liste wurde unter dem Namen Meßstellenliste abgelegt. Gesucht ist eine Funktion, die bei Angabe bei einer beliebigen Meßgröße aus der Meßstellenliste das Paar (Meßgröße, Wert) zurückgibt. Dazu könnte beispielsweise eine rekursive Funktion definiert werden:

```
(DEFUN assoc (meßgröße liste)
  (COND((NULL liste) NIL)
       ((EQ meßgröße (CAAR liste)) (CAR LISTE))
     (T  (assoc meßgröße (CDR liste))) ))
```

Erklärung:

IF Liste = leer THEN das Meßwertpaar ist nicht enthalten NIL und RETURN
IF Meßgröße = Meßgröße i THEN gib Meßwertpaar aus und RETURN
ELSE suche Meßgröße in der um das erste Element verkürzten Liste (rekursiver Aufruf von assoc)

(assoc meßgröße3 meßstellenliste) ⇒ (meßgröße3 wert3)

Bei Eingabe der aktuellen Parameter meßgröße3 und meßstellenliste würde das Paar (meßgröße3 wert3) zurückgegeben.
Die hier aufgezeigte Problemstellung kommt in ihrer verallgemeinerten Form häufig vor. Deshalb wurden in LISP Funktionen zur Behandlung sogenannter ASSOC-Listen fest implementiert. Allgemein hat eine ASSOC-Liste folgende Struktur:

((key1 wert1) (key2 wert2)...(keyn wertn))

Mit

(ASSOC item a-list &KEY TEST) ⇒ ASSOC-PAIR

kann obige Aufgabe gelöst werden. Die Angabe einer speziellen Testfunktion nach dem Schlüsselwort &KEY ist optional und ermöglicht außerdem Test auf physische Gleichheit (EQ) auch andere Tests, z. B. auf strukturelle Gleichheit mittels (EQUAL). Neben ASSOC zur Suche in ASSOC-Listen unterstützt LISP mit einer Reihe weiterer Funktionen die Arbeit mit ASSOC-Listen. Für die Aufnahme neuer Elemente in eine ASSOC-Liste gibt es die Funktion ACONS. Sie entspricht der CONS-Funktion bei gewöhnlichen Listen.

(ACONS key datum a-list) ⇒ new-a-list

ACONS nimmt also ein neues ASSOC-Paar in die ASSOC-Liste auf.

Beispiel:

```
(SETF meßstellenliste
  (Acons 'T-vorlauf 30 meßstellenliste))
(ASSOC 'T-vorlauf meßstellenliste) ⇒ '(T-vorlauf 30)
```

Bemerkt werden soll, daß ASSOC-Listen neben den speziell auf sie zugeschnittenen Funktionen natürlich auch mit allen allgemeinen Listenfunktionen behandelbar sind. Das zeigt auch das einführende Beispiel. Eine ASSOC-Liste ist also eine Liste, die aufgrund ihrer besonderen Struktur neben den Funktionen auf Listen noch eine Reihe spezieller Funktionen erlaubt. So könnte z. B. auch die Funktion DELETE aus dem Funktionsvorrat von LISP genutzt werden, die es ermöglicht, ein Element aus einer Liste wieder zu entfernen. Diese Funktion arbeitet destruktiv, d. h., das Element wird physisch aus der Liste entfernt. Im Gegensatz zu DELETE gibt es die Funktion REMOVE, die ein Element entfernt, ohne die Liste physisch zu verändern.
Damit kann das oben aufgenommene Element wieder aus der ASSOC-Liste entfernt werden.

(DELETE '(T-vorlauf 30) meßstellenliste : TEST 'EQUAL)

Man beachte: ":TEST 'EQUAL" erlaubt den Test auf strukturelle Gleichheit, also den Vergleich zweier Listen. Ohne diese Option könnte nur auf physische Gleichheit getestet werden, d. h., es könnte nur ein einfaches Element aus der Liste entfernt werden. Daraus folgt:

(DELETE 'd '(a b c d e f)) ⇒ (a b c e f).

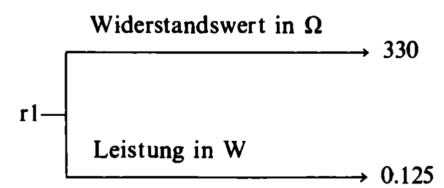
Die Arbeit mit ASSOC-Listen eröffnet umfangreiche Möglichkeiten, spezielle Strukturen der künstlichen Intelligenz zu realisieren. Mittels einer mehrfach verschachtelten ASSOC-Liste wurde in [1] eine Framestruktur aufgebaut. Dieses Beispiel findet sich im zum GC-LISP gehörenden Tutor.

2.4.3. Property-Listen

Bildeten ASSOC-Listen nur eine spezielle Form allgemeiner Listen, so sind Property-Listen als neuer Datentyp zu betrachten. Property-Listen sind weitestgehend mit Strukturen in C vergleichbar. Auch hier diene wieder ein *Beispiel* zur Einführung des neuen Listentyps: Für ein elektronisches Gerät soll eine Widerstandsliste erstellt werden. Diese Liste heiße W-liste und soll folgende Form haben:

```
(SETF W-liste)
  '(r1 r2 r3 ... rn)
```

Jeder Widerstand wird im Beispiel eindeutig durch zwei Kenngrößen, den Widerstandswert und die zulässige Leistung, beschrieben. Für r1 könnte also gelten:



Derartige Zuordnungen von Eigenschaften können in LISP vorteilhaft mit Property-Listen beschrieben werden. Die allgemeine Funktion zum Zugriff auf die Eigenschaften bildet:

(GET obj indicator ⇒ value.

Zum Setzen von Eigenschaften wird wie in den vergangenen Abschnitten die universelle Funktion SETF genutzt.

(SETF (GET obj indicator) value) ⇒ value

Im Beispiel müßte also wie folgt vorgegangen werden:

```
(SETF (GET 'r1 'widerstandswert) 330)
(SETF (GET 'r1 'leistung)        0.125)
```

Jetzt kann mit GET die Eigenschaft abgefragt werden:

```
(GET 'r1 'widerstandswert) ⇒  330
(GET 'r1 'leistung)        ⇒ 0.125
```

Durch die Funktion

(REMF obj indicator)

ist die Löschung einer gesetzten Eigenschaft möglich.

Aufgrund der Verwandschaft zu Strukturen in anderen Programmiersprachen ist die Bedeutung der Property-Liste offensichtlich. Erwähnt werden soll, daß der Wert einer Eigenschaft selbst wieder eine Liste sein kann. In anderen Programmiersprachen hat diese Möglichkeit ihre Entsprechung in der Verwendung von Zeigern. In LISP wird der Nutzer vollständig vom Umgang mit Zeigern befreit, was eine der wesentlichen Stärken von LISP ist. Es sei daran erinnert, daß in LISP beliebig komplexe Listenstrukturen behandelbar sind, was bei gewöhnlichen Programmiersprachen sehr schnell zu sehr unübersichtlichen Zeigerkonstruktionen führt.

In einem abschließenden Beispiel zu den Property-Listen soll nochmals die Eleganz rekursiver Funktionsdefinitionen demonstriert werden. Aus der o. g. Widerstandsliste sollen alle Widerstände ab einer bestimmten Leistung herausgesucht werden. Das Ergebnis ist in Form einer Liste zurückzugeben.

```
(DEFUN leistungsw (leistung liste ergebnisl)
  (COND ((NULL liste) ergebnisl)
        ((> =leistung (GET (CAR liste) 'leistung))
        (PUSH (CAR liste) ergebnisl)
        (leistungsw leistung (CDR liste) ergebnisl))
   (t   (leistungsw leistung (CDR liste) ergebnisl))) )
```

Erklärung:

IF Liste = leer THEN gib die in Ergebnisl gesammelten Widerstände aus und RETURN

IF Leistung des ersten Widerstandes der Liste > = Leistung

THEN nimm diesen Widerstand in die Ergebnisl auf und suche weiter nach entsprechenden Widerständen in der um das erste Element verkürzten Liste

ELSE suche weiter nach entsprechenden Widerständen in der um das erste Element verkürzten Liste (rekursiver Aufruf von leistungsw)

Eine Widerstandsliste mit Widerständen einer Leistung > = 0.5 W würde damit folgende Anfrage liefern:

(leistungsw 0.5 W-liste nil) ⇒ ergebnisl

(wird fortgesetzt)

F. Roß

PROLOG — eine Einführung für Automatisierungstechniker 5

4. Gleichheit in PROLOG

Wie im Abschnitt 3. gezeigt wurde, spielt bei der Abarbeitung eines PROLOG-Programms der Test auf Gleichheit zweier Terme eine besondere Rolle. Um die Gleichheit in PROLOG darzustellen, betrachte man den Ausdruck X=Y, wobei X und Y jeweils einen Term darstellen. Vier Fälle sind zu unterscheiden, wobei die Werte für X und Y vertauscht werden können:

a) X ist eine nichtinstanziierte (freie, ungebundene) Variable und Y ist ein Objekt (Konstante, Struktur, Liste, instanziierte Variable). Dieser Test ist immer erfolgreich, wobei X durch Y gebunden wird.

   Beispiele:

```
?—X=anlage_1.
X=anlage_1
Yes
?—zustand(anlage_1,frei)=X.
X=zustand(anlage_1,frei)
Yes
```

b) X und Y sind Konstanten bzw. durch Konstanten instanziierte Variablen. Dies ist der einfachste Fall des Tests auf Gleichheit. Er ist erfolgreich, wenn X und Y die gleiche Konstante darstellen, d. h., Integer und Atome sind zu sich selbst gleich.

   Beispiele:

```
?—15=15.
Yes
?—anlage_1=anlage_1.
Yes
?—anlage_1=anlage_2.
No.
```

   Besonders zu beachten ist der Unterschied der folgenden beiden Klauseln.

```
?—X=5,X=6.
?—X=5,X=5.
```

   In der ersten Klausel erfolgt in der ersten Teilaussage ein Test auf Gleichheit gemäß Fall a), d. h., X wird durch 5 gebunden.

Diese Variable ist in der zweiten Teilaussage also instanziiert. Dadurch erfolgt hier ein Test gemäß Fall b). Es wird die 6 mit der 5 verglichen, der Test scheitert und die Systemantwort lautet „No“. In der zweiten Klausel ist nach der Instanziierung von X durch 5 in der ersten Teilaussage auch die zweite Teilaussage erfolgreich und das System antwortet mit:

X=5
Yes

c) Die beiden Terme X und Y stellen jeweils eine Struktur dar. Der Test ist erfolgreich, wenn die Prädikatennamen und die Anzahl der Argumente übereinstimmen und wenn die korrespondierenden Argumente gleich sind.

Beispiele:

?—zustand(anlage_1,frei)=zustand(anlage_1,X).
X=frei
Yes
?—zustand(anlage_1,frei)=zustand(anlage_2,frei).
No

d) X und Y sind nichtinstanziierte Variablen. Der Test auf Gleichheit ist immer erfolgreich und im Ergebnis sind X und Y aneinander gebunden, d. h., wird im weiteren Verlauf der Programmabarbeitung eine der beiden Variablen mit einem Wert belegt, so erhält die andere ebenfalls diesen Wert.

Beispiele:

?—X=Y,X=anlage_1.
X=anlage_1
Y=anlage_1
Yes
?—X=Y,X=2,Y=5.
No

Im letzten Beispiel muß die Systemantwort „No“ erfolgen, da in der ersten Teilaussage X und Y aneinander gebunden werden, damit in der zweiten Teilaussage beide Variable mit dem Wert 2 instanziiert werden und in der dritten Teilaussage somit der Test 2=5 scheitert.

5. Listenverarbeitung

In diesem Abschnitt sollen die im Abschnitt 2.1.4. eingeführten Listen näher betrachtet werden. Die Liste ist in der nichtnumerischen Programmierung eine allgemeine Datenstruktur und somit auch in PROLOG. Listen lassen sich besonders gut zur Darstellung von Mengen (z. B. auch Lösungsmengen) verwenden, aber auch Graphen, Programme, Versorgungsnetze, Stadtpläne und vieles mehr können durch Listen dargestellt werden. Listen spielen in PROLOG zwar keine so große Rolle wie in LISP, sind jedoch auch hier vielseitig anwendbar.

Da in einer Liste nur auf das erste Element als einzelnes Element zugegriffen werden kann, erfordert dies eine andere Herangehensweise bei der Listenverarbeitung als in herkömmlichen Programmiersprachen. Grundsätzlich ist zu sagen, daß eine Liste untersucht wird, indem das erste Element untersucht wird und wenn der Test scheitert, wird die Restliste weiter untersucht. Um dies zu demonstrieren, sollen zwei häufig verwendete Listenoperationen in PROLOG-Klauseln umgesetzt werden.

5.1. „ist_Element“

Als erstes soll getestet werden, ob ein bestimmter Term Element einer Liste ist. Das Prädikat soll „ist_Element“ heißen und zwei Argumente, den zu testenden Term und die zu testende Liste, haben. Da PROLOG-Klauseln Regeln sind, wird zunächst die verbale Regel aufgestellt und diese dann in Prolog umgesetzt. Die verbale Regel lautet:

wenn der Listenkopf (erstes Element der Liste) gleich dem zu testenden Term ist,
dann ist dieser Term Element der Liste,
sonst soll die Restliste weiteruntersucht werden.

Die PROLOG-Klauseln dazu lauten:

ist_Element(X,[X¦_]).
ist_Element(X,[_¦Y]):—ist_Element(X,Y).

```
anhängen ([2¦[4]], [5,6],[2¦L3]):-anhängen([4],[5,6],L3).

anhängen ([4¦[]],[5,6],[4¦L3]):-anhängen([],[5,6],L3).

anhängen ([],[5,6],[5,6]).
```

Bild 13. Abarbeitung der Klausel „anhängen“

Dieses Beispiel macht das Typische der Listenverarbeitung deutlich. Nach einer Klausel, die die Testbedingung für das erste bzw. die ersten Element(e) enthält, steht eine weitere Klausel, die durch eine Rekursion die Suche in der Restliste fortsetzt. Die erste Klausel kann auch als Abbruchbedingung der Rekursion angesehen werden. Die beiden anonymen Variablen sagen aus, daß zum einen der Wert für die Restliste, zum anderen der Wert für das erste Listenelement nicht interessieren, da sie an dieser Stelle nicht für die Lösungssuche benötigt werden.

Folgender Dialog wäre nun möglich:

?—ist_Element(2,[anlage_1,2,4]).
Yes

5.2. „anhängen“

Als zweites Beispiel sollen Liste 1 und Liste 2 zu einer Gesamtliste verbunden, d. h., Liste 2 soll an Liste 1 angehängt werden. Die verbale Regel dazu lautet:

wenn die Liste 1 eine leere Liste ist,
dann ist die Liste 2 die Gesamtliste,
sonst ist das erste Element der Gesamtliste gleich dem ersten Element der Liste 1 und die Liste 2 wird an die Restliste der Liste 1 angehängt, wodurch die Restliste der Gesamtliste entsteht.

Das Prädikat soll „anhängen“ heißen und als Argumente die drei Listen haben:

anhängen ([ ],L2,L2).
anhängen ([X|L1],L2,[X|L3]):—anhängen (L1,L2,L3).

Auf die Anfrage

?—anhängen ([2,4],[5,6],X).

lautet die Antwort

X=[2,4,5,6]
Yes

Die einzelnen Inferenzschritte sind im Bild 13 dargestellt, wobei die Pfeile die Instanziierung der Variablen verdeutlichen sollen.

6. Arithmetik in PROLOG

Arithmetische Operationen in der Datenverarbeitung bedeuten Zahlenmanipulationen mehr oder weniger großen Umfangs. Da die Programmiersprache PROLOG für die Symbolverarbeitung geschaffen wurde, sind in ihr nur einfachste mathematische Operationen und nur die Verarbeitung von Integerzahlen möglich. Moderne PROLOG-Systeme (C-PROLOG, PROLOG-2, TURBO-PROLOG) erlauben zwar auch die Verarbeitung von Realzahlen und bieten einen umfangreichen Satz mathematischer Funktionen an, doch aufgrund der Abarbeitungsstrategie der Programmiersprache PROLOG (Inferenzmethode) werden mathematische Algorithmen in Klauselform stets langsamer abgearbeitet als in Form von Prozeduren herkömmlicher Programmiersprachen. Die Umsetzung eines mathematischen Algorithmus in TURBO-PROLOG und in TURBO-PASCAL ergab z. B. ein Verhältnis in der Abarbeitungsgeschwindigkeit von 12:1.

Müssen während der Abarbeitung eines PROLOG-Programms größere Zahlenmanipulationen vorgenommen werden, so sollten die dazu erforderlichen Algorithmen in eine herkömmliche Programmiersprache, wie FORTRAN oder PASCAL, umgesetzt werden und diese Prozeduren oder Programme in das PROLOG-Programm eingebaut werden.

(wird fortgesetzt)

XI. IMEKO-Weltkongreß, Teil 2[1)]

Meßtheorie und Meßtechnikausbildung

Sowohl die Meßtheorie als auch die Meßtechnikausbildung erhalten aus den Anforderungen, die an die Qualitätssicherung der industriellen Reproduktionsprozesse gestellt werden, neue Antriebe. Es ist bemerkenswert, daß aufgrund der hohen Geschwindigkeit des wissenschaftlich-technischen Fortschritts immer stärker gefordert wird, die Ausbildung im Sinne einer Vorbildung auf den Erwerb solider Grundlagenkenntnisse und die dazugehörigen Fähigkeiten und Fertigkeiten zu konzentrieren und die Spezialausbildung arbeitsplatzbezogen zu betreiben, wodurch die moderne Weiterbildung im Rang mit der früheren Ausbildung nahezu gleichzieht. Gerade die Instrument Society of America (ISA) verfügt hier über langjährige repräsentative Erfahrungen.
Meßinformationen sind in weiten Bereichen des gesellschaftlichen und individuellen Lebens unersetzliche Grundlage für richtige Entscheidungen und Handlungen. Die Meßtheorie systematisiert das akkumulierte Wissen über Meßprozesse insbesondere in Natur- und Technikwissenschaften sowie in der gesellschaftlichen Praxis und strebt nach seiner exakten Beschreibung sowie Verallgemeinerung. Die Meßtheorie schafft die Voraussetzungen für eine Erhöhung der Einheitlichkeit der Maße und Richtigkeit der Messungen. Durch Modellierung lassen sich Meßprozesse leichter beschreiben, voraussagen und in Mikrorechner implementieren. Modelle sind im Unterschied zu den Originalen (Meßsignalen, Meßsystemen, Meßprozessen) je nach Situation auch billiger, einfacher, raum- oder zeittransformiert, leicht variierbar, begrenzt optimierbar sowie einfacher zu lehren und zu lernen.
Für den Erkenntnisgewinn und die Theorienbildung ist von großem Gewinn, daß Originale sehr unterschiedlicher Natur zu isomorphen oder homomorphen Abbildungen führen können, die sich gleichen. Sie gestatten dann eine einheitliche Behandlung und Interpretation. In jedem Falle müssen die mit Hilfe eines Modells im Bildbereich gewonnenen Erkenntnisse im Objektbereich durch die Praxis bestätigt werden. Anderenfalls ist ihr Wert fraglich. Dabei ist zu beachten, daß es keine sogenannten „besten" Modelle gibt. Ein Modell ist gut, wenn es den Bedürfnissen der Praxis entspricht und mit den empirischen Erfahrungen aus der Praxis gut übereinstimmt.
Da jedes Modell in Abhängigkeit von seiner Zielorientierung unterschiedliche Seiten des Objekts hervorhebt oder unterdrückt, liefern zahlreiche unterschiedliche grobe Modelle häufig tiefergehende Einsichten in das Wesen des Objekts als ein einzelnes sehr verfeinertes Modell.
In Band 5 der Preprints wird eingegangen auf
- Strukturanalyse durch Bildverarbeitung und Normalmodellierung
- Echtzeitanalyse und Klassifikation von Oberflächendefekten
- modellbasierte Messungen im Zeit- und Frequenzbereich
- Multiprozessor-Systeme mit Selbsttest
- modellbasierte Überwachung und Diagnose in Fertigungsprozessen
- Einfluß moderner Instrumentierung auf die Meßtheorie
- elektrische Modelle für nichtelektrische Messungen
- Schaffung vereinheitlichter Meßskalen
- Selbstkalibrierung und Fehleranalyse
- Wechselverhältnis zwischen Meßgeräten und Meßtheorie
- rechnergestützte Meßgeräteprüfung und Kalibrierung
- integriertes Verfahren zur Signalanalyse
- Bildkorrektur in zerstörungsfreier Werkstoffprüfung
- mathematisches Modell für Messung von Flugzeuglärm.

Die Beiträge zur Meßtechnikausbildung betreffen Ausbildungsprogramme der Freien Universität Brüssel, Katholischen Universität Louvain, Ritsumeikan Universität Kyoto, Osaka Gas Company Ltd. Japan, Polytechnischen Hochschule von Wales, Norwegischen Hochschule für Technologie, Technischen Hochschule Leipzig, Technischen Universität Karl-Marx-Stadt.

Internationaler Dialog zur Meßtechnikausbildung

Im Zusammenhang mit der Meßtechnikausbildung ist weiterhin erwähnenswert, daß insbesondere unter dem Eindruck der stürmischen Qualitätsentwicklung in Japan seitens der USA ein internationaler Dialog der Wissenschaftlich-Technischen Gesellschaften (WTG) für Meßtechnik und Automatisierung angeregt wurde. Nach dem 1. Dialog 1982 in Philadelphia/USA, dem 2. Dialog 1984 in London und dem 3. Dialog 1986 in Beijing (Peking) fand der 4. Dialog am 15. Oktober 1988 in Houston, Texas (USA) statt. Die Leitung hatte der ISA-Präsident Dr. C. H. Cho. Weiterhin waren die ehemaligen ISA Präsidenten Wilson, Zoss, Bajek und Harvey sowie Vertreter der WTG aus China, BRD, Finnland, Frankreich, DDR, Indien, Italien, Japan, Schweden, England und UdSSR anwesend.
Die ISA stellte ihr beeindruckendes Instrumentation Training Program vor, das seit 1945 existiert und ständig aktualisiert und erweitert worden ist. Im 1989er Katalog sind über 100 Buchtitel angegeben, die von der ISA hauptsächlich nach 1984 herausgegeben wurden und von Sensoren über Meßsysteme bis zur Prozeßsteuerung mit MAP und TOP sowie Expertensystemen reichen.
Von einer neuen Buchserie „Resources in Measurement and Control (RMS)" wurde der erste Titel von T. A. Hughes: „Measurement and Control Basics" gezeigt und ausführlich erläutert. Mit 320 Seiten und 258 Illustrationen werden Prozeßsteuerung, Regelkreise, analoge und digitale Meßsysteme, Sensoren für Länge, Druck, Temperatur, Durchfluß, Feuchte, Dichte und analytische Messungen, Digitalrechner in der Prozeßsteuerung und programmierbare Regler behandelt. Übungen ergänzen das Buch.
Die Vertreter der ausländischen WTG wurden eingeladen, sich an dieser neuen Buchserie mit modernen Titeln zu beteiligen.
Die Leistungen der ISA umfassen weiterhin Kataloge, in denen Informationen über Hersteller und Geräte zusammengetragen und ständig ergänzt werden, Bücher mit ausgewählten Standards für die Meßtechnik (2 Bände, zusammen 1800 Seiten), über 70 spezielle Standardblattsammlungen, über 50 Videomappen mit durchschnittlich je 3 Videokassetten (Betamax und VHS, NTSC und PAL) Instrumentation Video Series IVS für das Selbststudium. Für denselben Zweck gibt es 20 Softwarepakete. Die Software läuft auf IBM-PC oder Kompatiblen.
Im Katalog ist ausgewiesen, daß die ISA von Oktober 1988 bis März 1989 im ISA Training Center etwa 70 zentralisierte 2- oder 4tägige Kurse mit Praktikum durchführt. Darüber hinaus bietet ISA sogenanntes In-Plant Training an und schafft alle Voraussetzungen dafür, daß die Arbeitskräfte arbeitsplatzbezogen in ihrem Betrieb weitergebildet werden. Im sogenannten Contract Training wird von der ISA weitgehend auf Sonderwünsche der Kunden eingegangen.
Sowohl während des 4. Internationalen Dialogs als auch bei mehreren anderen Anlässen wurde der 31. Dezember 1992 wie ein Menetekel behandelt. Der dann einheitliche westeuropäische Markt mit 320 Millionen Verbrauchern verlangt die Einführung des Internationalen Einheitensystems für Exporte aus den USA und fördert die Harmonisierung der Standards. Die Bestrebungen um die Vereinheitlichung des westeuropäischen Marktes werden aus amerikanischer Sicht als Modell für den zukünftigen Welthandel gesehen. In diesem Zusammenhang wurden während des Dialogs auch Fragen

[1)] Teil 1 erschien im H. 4/1989.

der gegenseitigen Anerkennung von Berufscharakteristika und Qualifizierungsnachweisen (Titeln) besprochen, ohne allerdings zu einem gemeinsamen Standpunkt zu kommen.

Strain Gage Golden Jubilee

Ein Höhepunkt des XI. IMEKO Weltkongresses in Houston waren das Jubiläumskolloquium und die festliche Jubiläumssitzung aus Anlaß der 50jährigen Wiederkehr der Erfindung von Dehnmeßstreifen, Kraftmeßdosen und Reißlack. Prof. P. K. STEIN, Präsident der *Stein Engineering Services, Inc.* ist es gelungen, Beiträge zur historischen Entwicklung der Kraftmessung aus 19 Ländern zu erhalten und in einem repräsentativen Festband der IMEKO STEIN, P. K.; KEMENY, T.; HAVRILLA, K. (eds.): „History of Strain Gages, Brittle Coatings and Load Cells — 50 Year Jubilee" (306 Seiten) zusammenzufassen. Die DDR ist mit einem Beitrag von Dr. N. SCHENKE, VEB Robotron Meßelektronik „Otto Schön" Dresden beteiligt.

ISA-Ausstellung Houston

Die Ausstellung der ISA stand ganz im Zeichen des weiteren Eindringens der rechnergestützten Informationsverarbeitung in die Meß- und Regelungstechnik. Mehrere Firmen hatten sich zur praktischen Vorführung der MAP-3.0-Konzeption zusammengeschlossen.
Die wichtigsten Argumente waren:
- **eine** Netzwerklösung
- **totale** Flexibilität
- **zukünftiges** Wachstumspotential
- **systemoffene** einheitliche Sprache.

Die meisten größeren Firmen versprechen in ihren Angeboten komplette fortgeschrittene Lösungen für die Prozeßsteuerung: Fortgeschrittene Überwachungs- und Datenverarbeitungssysteme SCADA (supervisory control and data acquisition) enthalten auch leistungsfähige Algorithmen für die statistische Prozeßsteuerung SPC (statistical process control).
Zur Verbesserung der Mensch-Prozeß-Kommunikation werden leistungsfähige Grafikpakete unter dem Motto: „Visualize your application programs!" angeboten.
Neue Entwicklungen bei der internen Digitalisierung von monolithischen Analog-Digital-Umsetzern erlauben eine neue Preisgestaltung. 16-bit-ADU kosten nunmehr soviel wie ehemals 12-bit-ADU kosteten. Die 12-bit-ADU sind unter 10 Dollar/Stück handelsüblich.
Speziell für die programmtechnische Unterstützung von universellen Personalcomputer-Instrumenten ist das Programmpaket „Measure" für IBM-PC und PS/2 sowie Kompatible auf den Markt gekommen. *National Instruments* wirbt mit der Handelsmarke TM „The Software is the Instrument".
Der auf der ISA-Ausstellung angebotene Laptop Computer GRID Case 1500 läuft auf Basis des 80C286 oder 80C386, ist batteriegestützt, wiegt 5 kg, hat die Abmessungen 38 × 29 × 6 ccm, enthält ein 640 × 400-Punkte-Display mit Rückbeleuchtung oder Plasmaschirm und kann als echter Feldrechner mit bis 40 Mbyte Festplattenspeicher und zahlreichen Interfaces ausgerüstet werden. Preis: 3500 Dollar bei Ausrüstung mit zwei Floppy-Disk-Laufwerken.

National Institute of Standards and Technology NIST

Mit Unterzeichnung des Omnibus Trade and Competitiveness Act am 23. August 1988 erhielt das weltbekannte *National Bureau of Standards NBS*, Gaithersburg MD den neuen verpflichtenden Namen *National Institute of Standards and Technology NIST.* Damit finden die erweiterten Verpflichtungen der Einrichtung gegenüber Staat und Wirtschaft auch in der Benennung eine adäquate Widerspiegelung.
Das umfangreiche Gelände unweit der Endhaltestelle der Nordwestlinie der U-Bahn von Washington D.C. umfaßt die Gebäude für

das National Measurement Laboratory
- Center for Basic Standards
- Center for Radiation Research
- Center for Chemical Physics
- Center for Analytical Chemistry
- Office of Measurement Services
- Office of Standard Reference Data

das National Engineering Laboratory
- Center for Manufacturing Engineering
- Center for Chemical Engineering
- Center for Fire Research
- Center for Building Technology
- Center for Applied Mathematics
- Center for Electronics and Electrical Engineering

Institute for Materials, Science and Engineering

Institute for Computer Sciences and Technology

Special Programs

Office of Research and Technology Applications

Office of Product Standards Policy

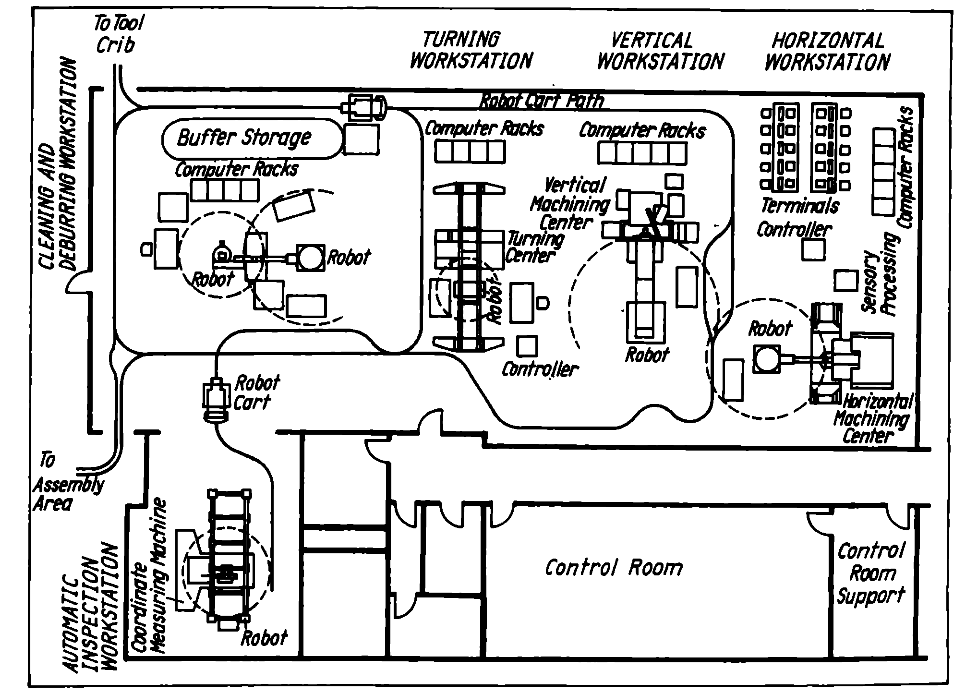
International Affairs

Resources

Directory.

Wichtigste Aufgabe der etwa 3000 Mitarbeiter, die über ein Jahresbudget von etwa 300 Millionen Dollar verfügen, besteht in aktiven Beiträgen zur Erhaltung der Wettbewerbsfähigkeit der amerikanischen Wirtschaft.
Die Verteilung der einzelnen Berufsgruppen sieht etwa wie folgt aus:
Ingenieure 30%, Rechentechniker 10%, Mathematiker 5%, Physiker 30%, Chemiker 20%, Andere 5%. Etwa 50% aller Mitarbeiter haben den akademischen Grad Ph. D. Zusätzlich sind jährlich etwa 200 Gastwissenschaftler aus etwa 40 Ländern am NIST tätig.
Früher wurden die erzielten Arbeitsergebnisse in demokratischer Gleichberechtigung allen Firmen zur Nutzung angeboten. Das hat sich nicht bewährt. Die Firmen haben sich gescheut, Geld auszugeben und ein schwer kalkulierbares Risiko zu übernehmen. Jetzt werden seitens des NIST exklusive Lizenzen an Firmen vergeben, wodurch seitens der Firmen viel mehr in die Überführung investiert wird.
Während die vom Autor besuchten Labors für Spannungsmessung, Längenmessung, Durchflußmessung und Kraftmessung eine im wesentlichen typische Instrumentierung aufwiesen, überraschte die Automated Manufacturing Research Facility AMRF (Bild 2)

Bild 2. Automated Manufacturing Research Facility des NIST



mit fünf untereinander verknüpften Fertigungszellen durch Art und Umfang der äußerst anspruchsvollen Ausstattung.

Zielstellung der AMRF ist die wissenschaftliche Untersuchung der industriellen Meßtechnik und Qualitätssicherung, des industriellen Datenmanagements, der Sensor- und Steuersysteme, der Maschinen- und Prozeßfähigkeit von Werkzeugmaschinen und Robotern, der Produktionsvorbereitung und Prozeßsicherheit, der Einsatzvorbereitung von Maschinen und Robotern sowie der Lagerung und des Transports bei flexibel automatisierter Produktion in kleinen Stückzahlen.

Bemerkenswert erscheint, daß dieses kooperative Projekt von Staat, Industrie, Armee und Universitäten erheblich von der U.S. Navy unterstützt wird und auch für über 5 Millionen Dollar Geräte von der Industrie erhielt. Etwa 40 Spezialisten aus der Industrie und mehr als 30 Mitarbeiter von 20 Universitäten sind in die NIST-Arbeitsgruppen integriert.

Der gegenwärtige Arbeitsstand reicht von CAD über CAM, CAT und CAQ bis CIM. CAD arbeitet mit IGES. Das Rechnernetz ist koaxial verkabelt. Neue QIA-Zielstellung (Quality in Automation) ist die wissenschaftliche Beantwortung der Frage, wie in der flexibel automatisierten Fabrik die Arbeitsproduktivität gesteigert und die Qualität gesichert werden können.

Ein weiterer Schwerpunkt sind die Schnittstellen. Es wurde erwähnt, daß etwa 90 Prozent der Hersteller von Kleinteilen in den USA, das sind etwa 100000 Firmen mit jeweils weniger als 50 Angestellten, für etwa 75% des gesamten USA-Handels mit Industriegütern gerade stehen müssen. Diese Firmen können sich nur langsam mit automatisierten Werkzeugmaschinen ausrüsten und stehen dann vor dem Problem, daß unterschiedliche Maschinen informatisch nicht zusammenpassen. Hier soll AMRF helfen.

Eine andere Aufgabe ist die wesentliche Vereinfachung der Programmierung von automatisierten Maschinen.

Am NIST laufen weiterhin systematische Untersuchungen zur Ultrapräzisionsbearbeitung mit Diamantdrehmaschinen. Die reproduzierbar erzielten Parameter für Positionierung, Geradheit und Ebenheit liegen bei 0,5 µm. Eine Automatisierung der Präzisionsbearbeitung und der Produktionsvorbereitung wird angestrebt. Zukünftige Zielrichtungen sind wissensbasierte Fertigungssysteme KMS (knowledge-based manufacturing systems).

Nationaler Qualitätspreis

Zur Förderung der Qualität in den USA wurde der *Malcolm Baldrige National Quality Award* gestiftet, der jedes Jahr vom Präsidenten der USA an die besten Hersteller, Dienstleistungsbetriebe und kleinen Geschäfte ausgegeben wird.

Beurteilungskriterien sind:

- Erfolg der oberen Leitungsebene bei der Entwicklung und Erreichung einer Qualitätskultur
- Effektivität der Sammlung von Daten für die Analyse der Qualitätslage der Einrichtung
- Effektivität der Integration der Qualitätsanforderungen in den Geschäftsplan
- Erfolg der Einrichtung bei der Mobilisierung des vollen Potentials der Arbeitskräfte für die Qualität
- Effektivität des Qualitätssicherungssystems für die Qualitätssteuerung aller Operationen
- Nachweis des Erfolgs von Qualitätssicherungsmaßnahmen anhand von quantitativ meßbaren Ergebnissen
- Effektivität bei der Ermittlung von Kundenwünschen und demonstrativer Erfolg bei deren Befriedigung.

Folgende Literaturstellen dienen als spezielle Mittel des NIST zur Unterstützung der Qualitätsarbeit:

TAYLOR, J. K.; OPPERMANN, V.: Handbook for the Quality Assurance of Metrological Measurements. NBS Handbook 145. Washington: U.S. Government Printing Office 1986.

URIANO, G. A.; GARNER, E. L.; KIRBY, R. K.; REED, W. P. (eds.): NBS Calibration Services Users Guide 1986—88/Revised. Washington: U.S. Government Printing Office 1987.

SEWARD, R. W. (ed.): NBS Standard Reference Materials Catalog 1988—89. Washington: U.S. Government Printing Office 1988.

Schlußfolgerungen

Der umsichtig organisierte und unter ausgezeichneten Randbedingungen erfolgreich durchgeführte XI. IMEKO-Weltkongreß hat zum internationalen Erfahrungsaustausch, Leistungsvergleich und nicht zuletzt auch zur gegenseitigen Verständigung aktiv beigetragen.

Es wurde erneut bewußt, daß die Meßtechnik weite Bereiche des gesellschaftlichen Lebens durchdringt, immer stärker zur Gewinnung neuer Erkenntnisse, zur Qualitätssicherung in der Produktion und Konsumtion, zur Bewahrung der Umwelt sowie zur Diagnose und Therapie in der Medizin herangezogen wird. Hochleistungsfähige Maschinen und Aggregate einschließlich Industrieroboter sowie flexible Automatisierung sind ohne eine leistungsfähige Meßtechnik undenkbar. Deshalb reicht die Bedeutung der Meßtechnik auch weit über ihr unmittelbares Produktionsvolumen in der Volkswirtschaft hinaus.

Vernachlässigungen in der Meßtechnik wegen ihres Anteils von weniger als 1% am Bruttosozialprodukt führen zu Folgekosten, die weit darüber liegen. Diese Abhängigkeit wird zunehmend bewußt. Soweit Theorie, Praxis und Intuition reichen, werden Meßtechnik, Automatisierungstechnik und Rechentechnik qualitativ erweitert, quantitativ verbessert und gemeinsam eingesetzt.

Auf dem Gebiet der intelligenten Meßtechnik und dem Entwurf sowie der Anwendung intelligenter transportabler Meßgeräte hat sich das Bild qualitativ gewandelt. Aktuelle Engstellen sind Prozeßkenntnis und Qualifizierung. Darüber hinaus haben sich Symptome der Destabilisierung der metrologischen Sicherheit von Messungen durch Einbeziehung umfangreicher ungeprüfter Programmtechnik (Software) verstärkt.

Es wurde bereits darauf aufmerksam gemacht, daß durch fehlerhafte oder unvollständig beschriebene Software Verständlichkeit, Überprüfbarkeit, Einheitlichkeit, Richtigkeit, Wiederholbarkeit und Vergleichbarkeit von Meßprozessen objektiv und subjektiv abnehmen. In Vorbereitung auf die XII. und XIII. IMEKO-Weltkongresse 1991 in Peking/VR China und 1994 in Turin/Italien werden sich die internationalen Anstrengungen zur Weiterentwicklung der Meßtechnik auf folgende Schwerpunkte konzentrieren:

Sensortechnik

- Erweiterung des Spektrums der Si- und GaAs-Sensoren
- Erweiterung der integrierten Optik
- Erweiterung des Spektrums der Fasersensoren
- technologische Verbesserung der konventionellen Sensoren
- Durchbruch bei intelligenten Sensoren
- Durchbruch bei optoelektronischen Sensoren für berührungslose Meß- und Prüfaufgaben
- Durchbruch in der Mikrometrologie für die Mikromechanik
- Durchbruch in der sprachlichen und bildlichen Mustererkennung

Gerätetechnik

- bedeutender Ausbau intelligenter Meßgeräte mit Selbstprüfung, Selbstkalibrierung, hohem Bedienkomfort und leichter Transportabilität
- CAD/CAM-kompatible intelligente Qualitätsmeßtechnik INQUAMESS
- Vereinheitlichung der Rechner
- Erweiterung der Korrelationsmeßtechnik, Ultraschallmeßtechnik, Analysenmeßtechnik, technischen Diagnostik, medizinischen Diagnostik, automatischen Bildauswertung, Produktionsmeßtechnik, Datenverarbeitung, Datenverdichtung und des Datenschutzes, der elektromagnetischen Verträglichkeit sowie Meßsignalfernübertragung

Programmtechnik

- Darlegung der Software
- Schaffung von strukturierter Software
- metrologische Prüfung von Software
- Anwendung standardisierter Software durch Angebot als hardwareimplementierte Firmware
- künstliche Intelligenz und Expertensysteme
- Selbstprogrammierung und adaptive Systeme

Schnittstellen

— international anerkannte Interfaces für Automatisierungsgeräte
— ergonomisch verbesserte Displays
— umfangreiche Verwendung von Softkeys und Touchscreens.

Als sicher kann gelten, daß die Kosten der Meßtechnik als Gerätekosten, Programmierkosten, Wartungskosten und Bedienkosten sowie die Fehlerfolgekosten als Wirkung durchgeführter oder unterlassener Messungen immer stärker ins Bewußtsein vordringen. Durch die ideenreiche Anwendung mathematischer und heuristischer Algorithmen und durch die massenhafte Sammlung von praktischen Erfahrungen im Umgang mit intelligenter Meßtechnik ist ein qualitativer Sprung bei der meßtechnischen Informationsgewinnung und dadurch auch bei der technologischen Beherrschung von Forschungs- und Produktionsprozessen international eingetreten.
Im Gefolge nehmen Qualitätsangebote und Qualitätsforderungen auf dem internationalen Markt gesetzmäßig weiter zu. Die Produktivität der Arbeit steigt an und wächst dabei schneller als die Produktion.
Die meßtechnische Beherrschung technischer Prozesse wird zu einem Schlüsselargument bei der Sicherung von Exportchancen.
Meßtechnische Urkunden von Staatsinstitutionen, Kalibrierdiensten und leistungsfähigen anerkannten Institutionen der Hochschulen und Industrie haben wachsende handelspolitische Bedeutung.
Teure Systemlösungen und billige (low-cost) Insellösungen werden international gleichrangig bearbeitet und angeboten. Weitere große Fortschritte sind im Angebot an sogenannten flexiblen kleinen Lösungen zu erwarten. Dabei werden klassische Meßgeräte durch Einbau von Einchiprechnern „intelligent" und kompatibel gemacht, oder es werden die Personalcomputer durch universelle Personalcomputer-Instrumente UPCI zu intelligenten Meßgeräten erweitert. Gegenwärtig sammeln die fortgeschrittenen Industrieländer in großer Breite Erfahrungen im Umgang mit intelligenten Meßgeräten. In den nächsten Jahren vollzieht sich auf der Basis dieser Erfahrungen ein einschneidender Wandel in der Instrumentierungstechnik. Neue Meßstrategien auf der Basis solider Prozeßkenntnisse gewinnen an Bedeutung. Die künftigen Erfahrungen basieren auf heute eingesetzten intelligenten Meßgeräten. Rückstände bei der Ausstattung mit intelligenten Meßgeräten führen zunehmend zur Verschlechterung der Wettbewerbsfähigkeit auf dem internationalen Markt. Durchgreifende Verbesserungen des Verhältnisses von Aufwand zu Nutzen, wesentliche Einsparungen an Werkstoff und Energie, Erhöhung der Qualität und Zuverlässigkeit der Erzeugnisse und nicht zuletzt ihre Konkurrenzfähigkeit und Devisenrentabilität auf dem Weltmarkt werden immer stärker von einer prozeßnahen Meß- und Prüftechnik abhängen. Zahlreiche hochproduktive moderne Technologien sowie prozeßtechnische Lösungen mit Industrierobotern sind ohne eine leistungsfähige Meßtechnik funktionslabil oder funktionsunfähig.
In wenigen Jahren können fehlende Applikationserfahrungen mit intelligenten Meßgeräten Qualitätslücken hervorrufen, die über alle bisherigen Erfahrungen weit hinausgehen. Die multiplikativ qualitätsmindernde Wirkung unterlassener Messungen führt gesetzmäßig zu volkswirtschaftlichen Verlusten, die bisher kaum kalkulierbare Dimensionen haben könnten. Der XI. IMEKO-Weltkongreß wies dagegen zahlreiche akzeptable Wege, um neue hohe Ziele erfolgreich anzusteuern.

msr 9032 Prof. Dr.-Ing. habil. D. HOFMANN, Mitglied des Generalrates der IMEKO

Buchbesprechungen

Drehzahlvariable Drehstromantriebe mit Asynchronmotoren. Von P.-K. BUDIG. Berlin: VEB Verlag Technik 1988. 296 Seiten, 210 Bilder, 38 Tafeln. 40,— M.
Bestell-Nr.: 553 888 3

Der Autor vermittelt in diesem Buch seine umfangreichen, durch eigene Forschungen gestützten Erfahrungen auf dem Gebiet der frequenzgesteuerten Asynchronantriebe. Schwerpunkt ist dabei das Zusammenwirken Stromrichter — Motor im stationären Betrieb. Fragen der Spannungs- und Stromoberschwingungen und der durch sie hervorgerufenen Drehmomentpendelungen und Zusatzverluste im Motor nehmen breiten Raum ein. Die Pulsmustersteuerung und verwandte Steuerverfahren des Wechselrichters werden ausführlich behandelt. Es werden Dimensionierungsrichtlinien angegeben, die insbesondere für den Entwurf spezieller Asynchronmaschinen oder für den Einsatz serienmäßiger Asynchronmaschinen für Wechselrichterspeisung wichtig sind.
Die Hauptabschnitte beinhalten:

— Vergleich der Gleichstrom- und Drehstromantriebstechnik, Energieökonomie
— Frequenz-/Spannungsgestellte Asynchronmaschinen
— Verfahren zur Erzeugung der Pulsmuster
— Stromrichtergespeister Asynchronmotor
— Moment der stromrichtergespeisten Asynchronmaschine
— Betrieb des Asynchronmotors am Wechselrichter
— Geregelter Antrieb mit Asynchronmotor
— Stromrichtergespeiste Asynchronmaschine.

Die umfassenden praktischen Erfahrungen des Autors und seine Fähigkeit, in großen Zusammenhängen zu denken, werden für den Leser des Buches zu jeder Zeit deutlich. Das Buch vermittelt Praktikern und Studierenden eine Fülle wertvoller Informationen.

B 2414 R. SCHÖNFELD

Regelungstechnik auf dem Personalcomputer. Von M. PESCHEL. Berlin: VEB Verlag Technik 1988. 206 Seiten. 22,— M.
Bestell-Nr.: 553 901 8

Es gab noch kein Buch von Prof. PESCHEL, das den Leser ohne Gewinn ließ. Worin und für welchen Leserkreis ist nun der Gewinn eines Buches zu suchen, dem der Autor selbst im Vorwort einen Zwittercharakter bescheinigt? Es besteht eigentlich aus vier durchgängig verknüpften Anliegen, die ihre inneren und äußeren Widersprüche nicht verleugnen:

— philosophisch-historisch-methodisch orientierte Betrachtungen hohen Allgemeinheitsgrades, mit Beispielen vom Makro- zum Mikrokosmos, aus Technik-, Bio-, Natur- und Gesellschaftswissenschaften
— eine Einführung in Modelle der Signal- und Systemtheorie ohne die traditionell vorherrschende Rolle der linearen Theorien
— zahlreiche Bezüge zu den in den letzten Jahren vom Verfasser betriebenen und stimulierten Forschungsgegenständen, insbesondere zur Dynamik der Lotka-Volterra-Gleichungen und zum EVOLON-Modell
— Programme mit Demonstrationscharakter für einen nicht genannten Personalcomputer, geschrieben in SIMONS BASIC, die auf keinem in der DDR produzierten Rechner ohne wesentliche Überarbeitung lauffähig sind.

Das ist ein großes Programm. Es setzt allerdings voraus, daß der Leser an jeder Stelle die Ziele des Autors kennt, mindestens erahnt, am besten aber schon ausgebildeter Fachmann auf einem die Systemtheorie tangierenden Gebiet ist. Diese Einschränkung ist viel weniger bedeutungsvoll, als man zunächst glauben möchte, denn wer vermag schon, die vier oben genannten Aspekte in ihrer Einheit zu erfassen? Für Lehrende (vielleicht weniger für Lernende) ist es eine Fundgrube. Leider dürften für fast alle Leser die Programme wenig Nutzen bringen (es sind immerhin 49 Programme auf etwa 60 Druckseiten), da wenigstens die Umorganisation der Prozeduren und die Anpassung der Grafik für einen Interessenten, dem z. B. ein Bildungscomputer, A 7100, EC 1834 zur Verfügung steht, fast einer Neuprogrammierung gleichkommt. Schwierig ist das nicht, aber 60 Seiten . . .!
Unbestritten bleibt das Verdienst, einen entscheidenden Schritt in Richtung zum com-

puterunterstützten Lehren und Lernen unter Nutzung eines aktuellen und interessanten Gegenstandes geschaffen zu haben. In PETER HACKS' „Columbus, oder: Die Weltidee zu Schiffe" lesen wir „Menschen von sehr genauem Verstand fällt es oft schwer, gemeinverständlich zu reden.". Tatsächlich ist es oft mühevoll, dem Assoziationengefüge beim Lesen zu folgen, die gelegentlich schwungvolle Freiheit in der Terminologie zu tolorieren. Aber es ist ein intellektuelles Vergnügen von hohen Graden, kein Koch-, aber ein Denkbuch, in dem manche beiläufige Bemerkung schon wieder die Quelle für ein neues Buch sein könnte.

B 2420 D. BOCHMANN

Fertigung, Prüfung und Montage von Automatisierungsanlagen mit Mikrorechnern. G. VENUS und H. BEUCHEL. Heft 18 der Schriftenreihe Automatisierungstechnik, Betriebssektion der KDT und Bereich Technologie des VEB Geräte- und Regler-Werke „Wilhelm Pieck" Teltow, 1987. 60 Seiten, 13 Bilder, 8,— M.

Anhand des mikrorechnergestützten Prozeßleitsystems audatec, das vom *VEB GRW „Wilhelm Pieck" Teltow* für die Automatisierung der verschiedensten verfahrenstechnischen Anwendungsfälle geschaffen wurde, gehen die Autoren auf die Schwerpunkte der Fertigung, Montage und Inbetriebnahme von bestimmten audatec-Komponenten ein.
Im Rahmen der technologischen Konzeption der stationären Fertigung werden eingehend der technologische Fertigungsumfang und -ablauf, die Fertigungsorganisation, die Gefäßfertigung und insbesondere die Prüfung von Baugruppen, Gefäßen und Anlagenteilen beschrieben. Die Montage von Automatisierungsanlagen, die den Fertigungsumfang und -ablauf sowie die Arbeitsvorschriften der Anlagenmontage umfassen, bilden den zweiten Schwerpunkt.
Maßnahmen der Inbetriebnahme von audatec-Automatisierungsanlagen sowie die Anforderungen an die Transport- und Lagertechnik werden kurz angedeutet. Es werden ausschließlich technische Gesichtspunkte des Produktionsprozesses von Automatisierungsanlagen dargestellt, wobei die Vernetzung mit vorbereitenden Prozessen, wie Materialwirtschaft bzw. mit betrieblichen Planungs- und Abrechnungsprozessen vernachlässigt wird.
Die Broschüre ist über die Betriebssektion der KDT oder über die Zentrale Informationsstelle Wissenschaft und Technik des *VEB GRW „Wilhelm Pieck" Teltow*, Oderstr. 74/76, Teltow, 1530, zu beziehen.

B 2379 L. BLACKERT

Betriebssysteme. Parallele Prozesse. 3., durchgesehene Auflage. Von H. ZIMA. Mannheim/Wien/Zürich: Bibliographisches Institut. 325 Seiten.

Im Buch, das offensichtlich auf Vorlesungen des Autors vor 1975 (neueste zitierte Literatur stammt von 1975) basiert, wird aufbauend auf einem mengentheoretischen Modell eines Betriebssystems für parallele Berechnungen ausführlich auf wesentliche Eigenschaften der Parallelität eingegangen: Synchronisation (dabei werden Sperrsynchronisation (wechselseitiger Anschluß) und Zustandssynchronisation unterschieden), Determiniertheit, maximale Parallelität und Verklemmungen (Deadlocks).
Besonders geht der Autor auf die dezentrale Verwaltung der Betriebsmittel ein und beschreibt ausführlich die wichtigsten Mechanismen zur Synchronisation: Semaphore, bedingte kritische Regionen, Monitore. Anhand zahlreicher Beispiele (als Algorithmen ausführlich abgehandelt) wird die Anwendung dieser Mechanismen für die Koordinierung paralleler Abläufe anschaulich erklärt. Zur Notation der Algorithmen wird eine formale Beschreibungssprache angewendet. Das Buch ist sehr sorgfältig und methodisch vorbildlich geschrieben. Es ist besonders für Studierende und Fachleute geeignet, die in ihrer Ausbildung nicht mit der Begriffswelt der Informatik konfrontiert wurden, sich dieser aber bedienen müssen. Das Studium des Werkes ist also trotz des rasanten Fortschritts bei der Hardware der Rechnersysteme und dem heutigen Entwicklungsstand der Softwaretechnologie, die seit dem Verfassen des Buches erreicht wurden, lohnenswert.

B 2328 P. NEUMANN

Mathematische Theorie des optimalen Experiments. Von S. M. ERMAKOV, A. A. ŽIGLJAVSKIJ. Moskau: Nauka 1987. 320 Seiten, 7 Bilder, 9 Tafeln.

In den letzten Jahren erfuhren die Arbeiten zur Theorie des Experiments einen breiten Aufschwung und fanden zahlreiche neue Anwendungsgebiete, insbesondere auch im Zusammenhang mit der Automatisierung wissenschaftlicher Experimente. Das vorliegende vom sowjetischen Hochschulministerium als Hochschullehrbuch für die Fachrichtung „Angewandte Mathematik" zugelassene Buch wendet sich an Studenten mathematischer und technischer Disziplinen, wobei neben guten Kenntnissen der Matrizentheorie, Wahrscheinlichkeitstheorie und mathematischen Statistik auch Kenntnisse auf dem Gebiet der mathematischen Optimierung verlangt werden. Über die traditionell in derartigen Büchern enthaltenen Abschnitte zur Parameterschätzung linearer Regressionsmodelle, zu Faktorplänen, zur klassischen Theorie der Versuchsplanung für lineare und nichtlineare Regressionsmodelle sowie zur Planung von Experimenten zur Extremwertermittlung und Hypothesenüberprüfung enthält das vorliegende Buch auch eigene Forschungsergebnisse der Autoren, die insbesondere in Abschnitten zur robusten oder unverzerrten Planung, zur Versuchsplanung in funktionalen Räumen und zur Planung von Simulationsexperimenten vorgestellt werden. Die Anwendung dieser Verfahren zur Versuchsplanung kann hohe ökonomische Effekte erbringen, setzt allerdings die Überwindung erheblicher Schwierigkeiten bei der Organisation und Durchführung der Experimente voraus.

B 2342 J.-A. MÜLLER

Explosionsgeschützte elektrotechnische Betriebsmittel. Von H.-J. LINSTRÖM u. a. Berlin: VEB Verlag Technik 1988. 95 Seiten, 34 Bilder, 7 Tafeln. 10,— M.
Bestell-Nr.: 553 891 2

Im Vorwort wenden sich die Autoren an alle, die mit dem Betreiben, Instandhalten und Reparieren von Ex-Betriebsmitteln zu tun haben, auch an Konstrukteure, Hersteller und Prüfstellen. Projektanten und Errichter explosionsgeschützter elektrotechnischer Anlagen sind zwar nicht benannt, aber ebenso interessiert. Zu explosionsgeschützten Betriebsmitteln im Sinne des Standards gehören nur solche mit standardisierter Kennzeichnung, die bei Explosionsgefährdung durch Gase und Dämpfe vorgeschrieben ist. Der Staubexplosionsschutz bleibt daher unberührt. Zur Einführung in die Thematik wird erläutert, wie eine Gasexplosion zustande kommt und um welche Wirkungen es geht. Ebenso ist zu erkennen, welche Rolle die energetisch und zeitlich unterschiedlichen Zündquellen dabei spielen und welche Klassifizierungen der Stoffe und der Betriebsmittel darauf eingehen. Der dominierende Abschnitt Schutzartkonzeptionen beschreibt die Ziele des gerätetechnischen Explosionsschutzes. Mit Bezug auf die RGW-konformen Erzeugnisstandards TGL 55037 ff. werden in logischer Folge allgemeine Konstruktionsgrundsätze, prinzipielle und konkrete Wirkungsweisen der Ex-Schutzarten und Besonderheiten detailliert dargestellt. Der 3. Abschnitt Beurteilungsergebnis der Arbeitsstätte und Zuordnung der elektrotechnischen Betriebsmittel erklärt die Prinzipien des primären und sekundären Explosionsschutzes, die Niveauunterschiede in der Sicherheit von Ex-Betriebsmitteln sowie die dementsprechende Zuordnung der Schutzarten zu den Gefährdungsgraden.
In den Abschnitten 4. und 5. kommen die Belange der Instandhaltung, Reparaturen und Änderungen zu Wort. Die Ordnung nach Schwerpunkten bei den vier wesentlichen Schutzarten d, e, p und i gibt dem Praktiker wertvolle Hinweise. Das gilt besonders für den Abschnitt 5. zur Reparatur, dessen eingängige Darstellungsweise die einschlägigen Rechtsvorschriften ausführlich kommentiert. Der letzte Abschnitt Hinweise für die Arbeit mit den Standards enthält ein Vorschriftenverzeichnis und gibt methodische Hinweise für Reparaturen. Mit einem knappen Literaturverzeichnis schließt die Broschüre ab.

B 2400 J. PESTER

Kurzfassungen

Pötschke, D.:

Künstliche Intelligenz und 5. Rechnergeneration

msr, Berlin 32 (1989) 5, S. 194—198

Im vorliegenden Beitrag wird auf einige Triebkräfte eingegangen, die das Entwicklungstempo, die Entwicklungsrichtung und das Investitionsvolumen der Forschungen zur künstlichen Intelligenz in den einzelnen Ländern bestimmen, insbesondere auf die verschiedenen nationalen und internationalen Projekte zur Entwicklung der 5. Rechnergeneration.

Uebel, W.; Friedrich, A.; Hoffmann, H.:

Bestimmung der Farbanteile mehrfarbiger Druckvorlagen mit bildanalytischen Mitteln

msr, Berlin 32 (1989) 5, S. 198—202

Für die automatische Farbgebungssteuerung im Mehrfarbendruckprozeß ist die Bestimmung der Farbanteile der Primärdruckfarben am gedruckten Bogen wesentlichste Voraussetzung. Auf der Grundlage des Neugebauer-Gleichungssystems können die Flächenbedeckungsgrade der Buntfarben durch fotoelektronische Messung der Farbvalenzen bestimmt werden. Für die Farbe Schwarz wird der Flächenbedeckungsgrad mit Bildanalysemethoden ermittelt, indem vom Meßsegment mit einer CCD-Flächenkamera 3 Farbauszugsbilder erzeugt werden.

Lunze, J.:

Wissensbasierte Beratung beim rechnergestützten Entwurf von Automatisierungssystemen

msr, Berlin 32 (1989) 5, S. 204—209

Es wird ein Beratungssystem als regelbasiertes System konzipiert, das den Entwurfsingenieur bei der Planung und Durchführung des rechnergestützten Entwurfes unterstützt. Das Beratungssystem übernimmt Teilaufgaben des Entwurfes, die bisher vom Entwurfsingenieur ausgeführt werden mußten. Es bietet dem Entwerfer in Abhängigkeit von der gestellten Entwurfsaufgabe Folgen von Entwurfsschritten an, durch die, von den vorhandenen Daten ausgehend, das Entwurfsziel erreicht werden kann.

Palm, R.:

Steuerung eines sensorgeführten Roboters unter Berücksichtigung eines unscharfen Reglerkonzepts

msr, Berlin 32 (1989) 5, S. 210—213

Die Regelung sensorgeführter Roboter erfordert wegen des nichtlinearen Charakters von Reaktionsprozessen zwischen Robotereffektor und Objekt den Einsatz robuster, leicht implementierbarer Regelungsalgorithmen. Es wird ein Fuzzy-Regler für Systeme 2. Ordnung vorgestellt, wobei die Phasenebene in 6 unscharfe Regionen geteilt wird. Für Fehler, Fehlergeschwindigkeit und Steuergrößen werden 12 Fuzzy-Mengen mit den korrespondierenden Zugehörigkeitsfunktionen gebildet. Fuzzy-Produktionsregeln führen zum unscharfen Steuervektor, anschließende Mittelwertbildung zum scharfen Steuerwert.

Iwainsky, A.:

Rechnergestützte Dokumentation in CAD-Prozessen

msr, Berlin 32 (1989) 5, S. 213—219

Im Interesse der Integration verschiedener CAD-Prozesse sowie der weiteren Aufwandsreduktion sind grafische Darstellungen möglichst hochgradig automatisiert aus denjenigen Produkt-, Prozeß- oder Projektmodellen abzuleiten, die sie lediglich illustrieren sollen. Der Beitrag konzentriert sich auf zwei Grundtypen grafischer Dokumentationen, bei denen diese Herausforderung besonders deutlich wird. Dabei handelt es sich um Werkstattzeichnungen und netzartige grafische Schemata.

Clauss, C.; Donath, U.; Kurth, G.; Schwarz, P.; Trappe, P.; Wittwer, H.:

KOSIM — ein Multi-Level-Simulator für den Schaltkreisentwurf

msr, Berlin 32 (1989) 5, S. 223—227

Unter Multi-Level-Simulation versteht man die Beschreibung von Teilen eines Systems auf unterschiedlichen Modellierungsniveaus und ihre gemeinsame Simulation mit Hilfe verschiedener Simulationsalgorithmen. In KOSIM sind Programme zur Logik- und Elektriksimulation gekoppelt. Das Programm gestattet die Simulation von Schaltungen mit maximal 32000 Elementen und 32000 Verbindungen. Die Implementierung erfolgte auf einem 32-bit-Rechner Robotron K1840. Das Programm konnte erfolgreich beim Entwurf von VLSI-Schaltkreisen eingesetzt werden.

Резюме

Pötschke, D.:

Искусственный интелекти и пятое поколение ЭВМ

msr, Berlin 32 (1989) 5, стр. 194—198

В данной статье отмечается ряд мотивов, определяющих темп развития, направление развития, а также объем капиталовложения по искусственному интеллекту в отдельных странах, в частности, отмечаются различные национальные и международные проекты по развитию пятого поколения ЭВМ.

Uebel, W.; Friedrich, A.; Hoffmann, H.:

Определение составляющих цветов многоцветных фотооригиналов при помощи фотоаналитических средств

msr, Berlin 32 (1989) 5, стр. 198—202

Для автоматического управления цветом в процессе многоцветной печати одним из самых важных предпосылок является определение составляющих цветов первичной печати на печатанном листе. На основе системы уравнений типа Неугебауэр могут быть определены степени покрытия поверхности многими цветами путем фотоэлектронного измерения цветов. Для черного цвета степень покрытия поверхности определяется методами анализа изображения таким образом, что от измерительного сегмента при помощи плоскостной камеры на основе ПЗС создаются 3 цветоделенных изображения.

Lunze, J.:

Консультация на знаний при проекте систем автоматизации при применении ЭВМ

msr, Berlin 32 (1989) 5, стр. 204—209

Конципирована система консультации как система на базе регулирования, которая помогает инженеру-проектанту при планировании и проведении проектов с помощью ЭВМ. Система консультации возьмет на себя частичные задачи проекта, которые до сих пор были выполнены проектантом. Эта система в зависимости от поставляемой задачи проекта предлагает разработчику последовательности шагов проекта, при помощи которых, исходя из имеющихся данных, может быть достигнута цель проекта.

Palm, R.:

Управление роботом при помощи датчика с учетом неточного концепта регулятора

msr, Berlin 32 (1989) 5, стр. 210—213

Регулирование управляемых сенсорами роботов требует из-зна нелинейного характера процессов реакции между эффектором робота и объектом применения солидных, легко имплементируемых алгоритмов регулирования. Представлен регулятор типа Фуцци для систем второго порядка, причем плоскость фаз подразделяется на 6 нечетких областей. Для погрешностей, скорости погрешностей и величин управления образованы 12 количеств типа Фуцци с корреспондирующими функциями принадлежности. Правила производства типа Фуцци приводят к нечеткому вектору управления, к последующему образованию среднего значения к четкому значению управления.

Iwainsky, A.:

Документация при помощи ЭВМ в процессах автоматизации проектирования

msr, Berlin 32 (1989) 5, стр. 213—219

В интересах интеграции различных процессов автоматизации проектирования, а также дальнейшего уменьшения затрат необходимо выводить графические изображения по возможности с высокой степенью автоматизации из тех моделей продукта, процесса или проекта, которые они должны лишь иллюстрировать. В статье отмечены два основных типа графической документации, при которых данные требования особенно ясны. Это касается рабочих чертежей и графических схем в виде сетей.

Clauss, C.; Donath, U.; Kurth, G.; Schwarz, P.; Trappe, P.; Wittwer, H.:

КОЗИМ — симулятор с многими уровнями для разработки схем

msr, Berlin 32 (1989) 5, стр. 223—227

Под симулацией с многими уровнями понимают описание частей одной системы на различных уровнях моделирования и их совместная симуляция с помощью различных алгоритмов симуляции. В КОЗИМ связаны программы по логической и электрической симуляции. Программа позволяет симуляцию схем с элементами не более 32000 и соединениями не более 32000. Имплементация производится на 32-хразрядной ЭВМ РОБОТРОН K1840. Программа успешно могла быть использована при разработке схем сверхвысокой интеграции.

Summaries

PÖTSCHKE, D.:

Artificial Intelligence and Fifth Computer Generation

msr, Berlin 32 (1989) 5, pp. 194—198

The author deals with some motive forces which determine the speed and direction of development as well as the investment extent of the research for artificial intelligence in the individual countries, the different national and international projects as to the development of the fifth computer generation being particularly treated.

UEBEL, W.; FRIEDRICH, A.; HOFFMANN, H.:

Determination of the Colour Portions of Multicoloured Printing Models by Means of Image Analysis

msr, Berlin (1989) 5, pp. 198—202

For automatic colouration control in the multicolour printing process, the determination of the colour portions of the primary printing colours in the printed sheet represents the most essential prerequisite. On the basis of the Neugebauer equation system, the degrees of area coverage of the natural colours can be determined by photoelectronic measurement of the colour valences. For the colour black the degree of area coverage is obtained by means of image analysis methods generating three colour extraction images from the measurement segment by a CCD area camera.

LUNZE, J.:

Knowledge-Based Consultation with the Computer-Aided Design of Automation Systems

msr, Berlin 32 (1989) 5, pp. 204—209

The author configures a consultation system as a rule-based system which is to support the automation engineer in planning and execution of the computer-aided design. The consulting system takes over partial design problems which have been carried out by the automation engineer up to now. In dependence upon the design problem posed it offers the designer sequences of design steps leading from the available data to the design object.

PALM, R.:

Control of a Sensor-Guided Robot with Consideration of a Fuzzy Controller Concept

msr, Berlin 32 (1989) 5, pp. 210—213

Due to the nonlinear character of the mutual reaction processes between robot effector and object, the feedback control of sensor-guided robots requires the use of robust control algorithms easily implementable. The author presents a fuzzy feedback controller for second-order systems, the phase plane being divided into six fuzzy regions. For errors, error speed and control quantities, 12 fuzzy sets together with the corresponding membership functions are formed. Fuzzy production rules lead to the fuzzy control vector, subsequent averaging producing the nonfuzzy control value.

IWAINSKY, A.:

Computer-Aided Documentation in CAD Processes

msr, Berlin 32 (1989) 5, pp. 213—219

With regard to the integration of different CAD processes and the further reduction of expense graphical representations should be derived in a highly automated way just from that product, process and project models which are to be only illustrated by them. The author treats two basic types of graphical documents making these challenges particularly evident. The documents in question are workshop drawings and network-like graphical schemes.

CLAUSS, C.; DONATH, U.; KURTH, G.; SCHWARZ, P.; TRAPPE, P.; WITTWER, H.:

KOSIM — A Multi-Level Simulator for Integrated Circuit Design

msr, Berlin 32 (1989) 5, pp. 223—227

In the multi-level simulation parts of a system are described on different modelling levels and commonly simulated by means of different simulation algorithms. In KOSIM, programs for logical and electrical network simulation are coupled one with the other. The program allows the simulation of circuits up to 32,000 elements and 32,000 connections. It is implemented on a 32-bit Robotron K1840 computer and could be applied successfully to VLSI circuit design.

Résumés

PÖTSCHKE, D.:

Intelligence artificielle et la cinquième génération des ordinateurs

msr, Berlin 32 (1989) 5, pp. 194—198

L'auteur s'occupe de quelques ressorts qui déterminent la rapidité et la direction du développement ainsi que l'étendue d'investissement pour la recherche en intelligence artificielle dans les individuels pays, les différents projets nationaux et internationaux pour le développement de la cinquième génération des ordinateurs étant particulièrement taités.

UEBEL, W.; FRIEDRICH, A.; HOFFMANN, H.:

Détermination des portions de couleur des modèles d'impression multicolores à l'aide des méthodes de l'analyse des images

msr, Berlin 32 (1989) 5, pp. 198—202

Pour la commande automatique de la coloration dans le processus de chromotypographie, la détermination des portions de couleurs des encres d'impression primaires dans la feuille imprimée représente la condition préalable la plus essentielle. Sur la base du système d'équations de Neugebauer, on peut déterminer les degrés de couverture superficielle des couleurs naturelles par une mesure photo-électronique des valences de couleur. Pour la couleur noir, le degré de couverture superficielle est obtenu par les méthodes de l'analyse des images où trois images d'extraction de couleur sont engendrées à partir du segment de mesure par une caméra superficielle CCD.

LUNZE, J.:

Consultation basée sur connaissance dans la conception assistée par ordinateur des systèmes d'automatisation

msr, Berlin 32 (1989) 5, pp. 204—209

L'auteur configure un système conseilleur comme système basé sur règles qui doit supporter l'automaticien dans la planification et réalisation des projets assistés par ordinateur. Le système conseilleur assume des tâches partielles de la conception qui avaient été exécutées par l'automaticien jusqu' à présent. En dépendence du problème de projet posé, il offre à l'ingénieur des séquences de pas de projet qui conduisent à partir des données disponibles au but de projet.

PALM, R.:

Commande d'un robot guidé par capteur en considération d'un concept de régulateur flou

msr, Berlin 32 (1989) 5, pp. 210—213

A cause du caractère non linéaire des processus de réaction mutuelle entre l'effecteur de robot et l'objet, il faut employer des algorithmes de réglage robustes et facilement implémentables. L'auteur présente un régulateur flou pour les systèmes du deuxième ordre, le plan de phase étant divisé en six régions floues. Pour erreur, vitesse d'erreur et grandeurs de commande, on forme 12 ensembles flous et les correspondantes fonctions d'appartenance. Les règles de production floues conduisent à un vecteur de commande flou, la formation subséquente da la valeur moyenne produisant la valeur non floue de commande.

IWAINSKY, A.:

Documentation assistée par ordinateur dans les processus CAO

msr, Berlin 32 (1989) 5, pp. 213—219

En vue de l'intégration de différents processus CAO ainsi qu'une dépense plus réduite, il faut dériver les représentations graphiques d'une manière hautement automatisée à partir de tels modèles de produit, processus et projet qui ne sont qu'illustrés par elles. L'auteur traite deux types de base des documentations graphiques par lesquels ces défis deviennent très évidents. Il s'agit de dessins d'atelier et de schémas graphiques en forme de réseaux.

CLAUSS, C.; DONATH, U.; KURTH, G.; SCHWARZ, P.; TRAPPE, P.; WITTWER, H.:

KOSIM — un simulateur à plusieurs niveaux pour la conception des circuits intégrés

msr, Berlin 32 (1989) 5, pp. 223—227

En simulation à plusieurs niveaux, des parties d'un système sont décrites sur de différents niveaux de modélisation et simulées en commun à l'aide de différents algorithmes de simulation. En KOSIM, des programmes pour la simulation de la logique et pour la simulation des réseaux électriques sont couplés les uns avec les autres. KOSIM permet de simuler des circuits contenant 32,000 éléments et 32,000 connexions au maximum. Il estimplémenté sur une calculatrice à 32 bits Robotron K1840 et était appliqué avec succès à la conception des circuits VLSI.

Neue Bücher aus der UdSSR

Hierbei handelt es sich um neue Titel aus der UdSSR, die dem sowjetischen Neuerscheinungsdienst „Novye knigi" entnommen sind. Die Bücher können über den Volksbuchhandel bestellt werden.

[1] Bondarenko, V.; Achmetov, B.: Logiko-električeskoe modelirovanie bol'šich sistem (Logisch-elektrische Modellierung großer Schaltungen). Fan (UzSSR). 3. Quart. 1989. NK 88-39 (43).

[2] Kabulov, V.: Algoritmizacija v social'no-ekonomičeskich sistemach (Algorithmierung in sozialökonomischen Systemen). Fan(UzSSR). 4. Quart. 1989. NK 88-39 (44).

[3] Rašidov, Ju.: Optimal'noe upravlenie stochastičeskimi sistemami so svobodnym konečnym vremenem (Optimale Steuerung stochastischer Systeme mit freier Endzeit). Fan(UzSSR). 3. Quart. 1989. NK 88-39 (45).

[4] Sadykov, S.; Kan, V.; Samandarov, I.: Metody vydelenija strukturnych priznakov izobraženij (Methoden zur Aussonderung struktureller Kennzeichen von Abbildungen). Fan(UzSSR). 3. Quart. 1989. NK 88-39 (46).

[5] Aimbetov, N.; Zaitov, I.; Točilin, V.: Točnost' optimizacionnych rasčetov (Genauigkeit von Optimierungsberechnungen). Fan(UzSSR). 4. Quart. 1989. NK 88-39 (50).

[6] Magrupov, T.: Grafy, seti, algoritmy i ich priloženija (Graphen, Netze, Algorithmen und ihre Anwendungen). Fan(UzSSR). 4. Quart. 1989. NK 88-39 (53).

[7] Makarov, G.; Umarov, N.: Zony dostovernogo obnaruženija sobytij v sistemach avtomatičeskogo kontrolja (Zonen der zuverlässigen Erkennung von Ereignissen in Systemen der automatischen Kontrolle). Fan(UzSSR). 3. Quart. 1989. NK 88-39 (134).

[8] Zachidov, B.; Ajupov, R.: Algoritmy modelirovanija i upravlenija technologičeskimi processami (Algorithmen der Modellierung und Steuerung technologischer Prozesse). Fan(UzSSR). 4. Quart. 1989. NK 88-39 (136).

[9] Kadyrov, A.: Mašinnye metody modelirovanija i issledovanija strukturno-složnych sistem (Maschinelle Methoden zur Modellierung und Untersuchung strukturell-komplizierter Systeme). Fan-(UzSSR). 3. Quart. 1989. NK 88-39 (137).

[10] Pajziev, E.: Informacionno-strukturnye modeli upravlenija električeskimi setjami (Informationell-strukturelle Modelle der Steuerung elektrischer Netze). Fan(UzSSR). 4. Quart. 1989. NK 88-39 (138).

[11] German, E.: Modelirovanie operacij v organizacionnych sistemach (Modellierung von Operationen in Organisationssystemen). Izd-vo Tomsk. un-ta. 4. Quart. 1989. NK 88-40 (38).

[12] Il'in, V.: Algoritmy planirovanija povedenija integral'nych robotov v uslovijach nepolnoj informacii o strukture sredy (Algorithmen zur Planung des Verhaltens integraler Roboter bei unvollständiger Information über die Struktur). Izd-vo Tomsk. un-ta. 2. Quart. 1989. NK 88-40 (39).

[13] Tonkonogov, Ju.: Poisk dvižuščegosja signala v mnogokanal'noj sisteme (Suche eines beweglichen Signals im mehrkanaligen System). Izd-vo Tomsk. un-ta. 4. Quart. 1989. NK 88-40 (41).

[14] Trovoženko, B.: Vydelenie trendov vremennych rjadov i potokov sobytij (Aussonderung von Zeitreihentrends und aus Ereignisströmen) Izd-vo Tomsk. un-ta. 3. Quart. 1989. NK 88-40 (42).

[15] Šurygin, Ju. u. a.: Proektirovanie ustrojstv vyčislitel'noj techniki (Projektierung von Anlagen der Rechentechnik). Izd-vo Tomsk. un-ta. 4. Quart. 1989. NK 88-40 (70).

[16] Kučin, B.; Altunin, A.: Avtomatizirovannye informacionnye sistemy ob'ektov gazosnabženija (Automatisierte Informationssysteme von Objekten der Gasversorgung). Nedra. 4. Quart. 1989. NK 88-40 (76).

[17] Bimatov, V.: Avtomatizirovannye bazy dannych (Automatisierte Datenbasen). Izd-vo Tomsk. un-ta 1. Quart. 1989. NK 88-40 (297).

[18] Avtomatizacija issledovanija i proektirovanija sistem upravlenija (Automatisierung der Untersuchung und Projektierung von Steuerungssystemen). Ilim (KirgSSR). 1. Quart. 1989. NK 88-43 (26).

[19] Krivenko, V.: Kvaziekvivalentnye preobrazovanija optimizacionnych modelej v zadačach proektirovanija ASUTP (Quasiäquivalente Transformation von Optimierungsmodellen in Aufgaben der Projektierung von ASUTP). Ilim (KirgSSSR). 2. Quart. 1989. NK 88-43 (29).

[20] Šaršenaliev, Ž.; Kalmanbetov, M.: Dekompozicija sistem s malymi parametrami (Dekomposition von Systemen mit kleinen Parametern). Ilim (Kirg SSR). 2. Quart. 1989. NK 88-43 (29).

[21] Džanybekov, Č.: Modelirovanie gidrogeodinamičeskich processov s primeneniem. EVM (Modellierung von hydrogeodynamischen Prozessen unter Anwendung der EDV-Anlagen). Ilim (KirgSSR). 4. Quart. 1989. NK 88-43 (36).

[22] Avtomatičeskoe upravlenie processom vodoraspredelenija (Automatische Steuerung von Prozessen der Wasserverteilung). Ilim (KirgSSR). 2. Quart. 1989. NK 88-43 (106).

[23] Matematičeskoe modelirovanie. Metody opisanija i issledovanija složnych sistem (Mathematische Modellierung. Methoden zur Beschreibung und Untersuchung großer Systeme). Nauka. 4. Quart. 1989. NK 88-44 (73).

[24] Cyrdja, F.: Informatika, socialnoe poznanie, upravlenie. (Informatik, soziale Erkenntnis, Steuerung). Ştiinca (MoldSSR). NK 88-44 (76).

[25] Gonca, M.; German, L.; Prisakaru, K. u. a.: Issledovanija po prikladnoj matematike i informatike (Untersuchungen zur angewandten Mathematik und Informatik). Ştiinca (MoldSSR). NK 88-44 (82).

[26] Gončarjuk, N.: Matematika i informatika v električeskich setjach (Mathematik und Informatik in elektrischen Netzen). Ştiinca (MoldSSR). 1. Quart. 1990. NK 88-44 (83).

[27] Sekrieru, I.: Čislennoe rešenije matematičeskich zadač (Numerische Lösung mathematischer Aufgaben). Stiinca (MoldSSR). NK 88-44 (87).

[28] Levin, M.: Programmnoe obespečenie dlja naučno-techničeskich rasčetov na SM EVM (Software für wissenschaftlich-technische Berechnungen auf Rechnern vom Typ SM). Ştiinca (MoldSSR). 3. Quart. 1989. NK 88-44 (149).

[29] Malpas, Dž.: Reljacionnyj jazyk PROLOG i ego primenenie (Relationale Sprache PROLOG und ihre Anwendung. Übers. aus dem Engl.). Nauka. 4. Quart. 1989. NK 88-44 (150).

msr 9049

Impressum

messen · steuern · regeln

Herausgeber: Kammer der Technik, Wissenschaftlich-Technische Gesellschaft für Meß- und Automatisierungstechnik (WGMA)

Verlag: VEB Verlag Technik, Oranienburger Str. 13/14, PSF 201, Berlin, DDR-1020, Telegrammadresse: Technikverlag Berlin; Telefon 2 87 00; Telex: 011 222 8 techn dd

Verlagsdirektor: Dipl.-Ing. Klaus Hieronimus

Redaktion: Dr.-Ing. Dietrich Werner, Verantwortlicher Redakteur (Telefon: 2 87 03 62)

Dr.-Ing. Jürgen Wede, Redakteur (Telefon: 2 87 03 74)

Gestaltung: Veronika Becker

Lizenz-Nr.: 1112 des Presseamtes beim Vorsitzenden des Ministerrates der Deutschen Demokratischen Republik

AN (EDV): 15 936

Erscheinungsweise: monatlich 1 Heft

Heftpreis: 4,— M, Abonnementpreis vierteljährlich 12,— M; Auslandspreise sind den Zeitschriftenkatalogen des Außenhandelsbetriebs BUCHEXPORT zu entnehmen.

Gesamtherstellung: VEB Druckerei „Thomas Müntzer", Bad Langensalza, 5820

Anzeigenannahme: Für Bevölkerungsanzeigen alle Anzeigen-Annahmestellen in der DDR, für Wirtschaftsanzeigen der VEB Verlag Technik, Oranienburger Str. 13/14, PSF 201, Berlin, 1020. Anzeigenpreisliste Nr. 7; Auslandsanzeigen: Interwerbung GmbH, Hermann-Duncker-Str. 89, Berlin, DDR-1157

Bezugsmöglichkeiten:

DDR: sämtliche Postämter

SVR Albanien: Direktorije Qendrore e Perhapjes dhe Propaganditit te Librit Bruga Konference e Pezes, Tirana

VR Bulgarien: Direkzia R.E.P., 11 a, Rue Paris, Sofia

VR China: China National Publications Import and Export Corporation, West Europe Department, P.O. Box 88, Beijing

ČSSR: PNS — Ústřední Expedicia a Dovoz Tisku Praha, Slezska 11, 12000 Praha 2; PNS, Ústredná Expedicia a Dovoz Tlaće, Posta 002 885 47 Bratislava

SFR Jugolawien: Jugoslovenska Knjiga, Terazije 27, Beograd; Izdavačko Knjižarako Produzeče MLADOST, Ilica 30, Zagreb

Koreanische DVR: CHULPANMUL Korea Publications Export & Import Corporation, Pyongyang

Republik Kuba: Empresa de Comercio Exterior de Publicaciones, O'Reilly No. 407, Ciudad Habana

VR Polen: C. K. P. i. W. Ruch, Towarowa 28, 00-958 Warszawa

SR Rumänien: D.E.P. Bucureşti, Piaţa Scienteii, Bucureşti

Ungarische VR: P.K.H.I., Külföldi Előfizetési Osztály, P.O. Box 16, 1426 Budapest

UdSSR: Städtische Abteilungen von Sojuzpečat oder Postämter und Postkontore

SR Vietnam: XUNHASABA, 32, Hai Ba Trung, Hanoi

BRD und Berlin (West): ESKABE Kommissions-Grossobuchhandlung, Postfach 36, 8222 Ruhpolding/Obb.; Helios Literatur-Vertriebs-GmbH, Eichborndamm 141—167, Berlin (West) 52; Kunst und Wissen Erich Bieber OHG, Postfach 46, 7000 Stuttgart 1; Gebrüder Petermann, BUCH + ZEITUNG INTERNATIONAL, Kurfürstenstraße 111, Berlin (West) 30

Österreich: Helios Literatur-Vertriebs-GmbH & Co. KG, Industriestraße B 13, A-2345 Brunn am Gebirge

Schweiz: Verlagsauslieferung Wissenschaft der Freihofer AG, Weinbergstraße 109, 8033 Zürich

Alle anderen Länder: örtlicher Buchhandel; BUCHEXPORT Volkseigener Außenhandelsbetrieb der Deutschen Demokratischen Republik, Postfach 160, Leipzig, DDR-7010 und Leipzig Book Service, Talstraße 29, Leipzig, DDR-7010.

Abnahmebedingungen:

Die Redaktion veröffentlicht nur Originalbeiträge. Für Nachdrucke, Übersetzungen oder Vervielfältigungen jeglicher Art gilt der Rechtsvorbehalt. Die Redaktion behält sich Änderungsvorschläge bei der Beitragsbearbeitung vor. Die Beiträge müssen generell mit einer Veröffentlichungsfreigabe des Betriebes oder der Institution der Autoren versehen sein.

Bei der Anfertigung von Beiträgen sind unbedingt die Manuskripthinweise zu beachten. Diese sind dem Heft 5/1988, S. 237, zu entnehmen bzw. werden von der Redaktion auf Anfrage zugesandt. Die Einhaltung der Manuskripthinweise bildet die Voraussetzung für eine zügige und sachgerechte Beitragsbearbeitung.

6. Internationale Wissenschaftliche Konferenz „Anlagenautomatisierung“

Die Technische Hochschule Leipzig führt vom 22. bis 24. Mai 1990 o. g. Konferenz durch.
In den Vorträgen dieser Konferenz werden wissenschaftliche Ergebnisse und technische Entwicklungen behandelt, die dem fachlichen Profil der Sektion Automatisierungsanlagen entsprechen.
Schwerpunkte:

Rechnergestützte Projektierung von Automatisierungsanlagen

- CAD für Aufgabenaufbereitung und -prüfung
- CAD für Ausführungsprojektierung
- Softwareprojektierung
- Künstliche Intelligenz in der Projektierung
- Mensch-Maschine-Kommunikation
- Fachsprachen, Datenbanken
- Anwendungen

Prozeßautomatisierung

- Identifikation, Modellierung, Simulation
- CAD für automatische Steuerungen
- Theorie automatischer Steuerungen
- Adaptive und robuste Systeme
- Softwaretechnologie
- Expert-Control
- Anwendungen

Funktionseinheiten der Automatisierungsanlagen

- Intelligente Meß- und Stelltechnik
- Sensortechnik
- Signalanalyse
- Kommunikationssysteme
- Prozeßleitsysteme
- Anwendungen

Experten- und Beratungssysteme in der Automatisierungstechnik

- Systemtheoretische Grundlagen
- Technologie wissensbasierter Systeme
- KI (Künstliche Intelligenz)-Werkzeuge
- Wissenspräsentation
- Anwendungen.

Anfragen richten Sie bitte an:

Technische Hochschule Leipzig
Sektion Automatisierungsanlagen
6. Internationale Wissenschaftliche Konferenz
Karl-Liebknecht-Straße 132
Leipzig
7030

Systems Science X

Vom 19. bis 22. September 1989 findet in Wroclaw/VR Polen die o. g. internationale Konferenz zur Systemwissenschaft statt. Seit 1980 wird diese Veranstaltung umlaufend von der TU Wroclaw/VR Polen, dem Coventry Lanchester Polytechnic/England und der Wright State University/USA veranstaltet.
In diesem Jahr organisiert das „Institute of Control and Systems Engineering“ der TU Wroclaw die Konferenz.
Schwerpunkte:

- Allgemeine System- und Steuerungstheorie
- Systemidentifikation, -modellierung und -simulation
- Systemoptimierung
- Steuerung großer Systeme
- Anwendung von Mikrorechnern und Computernetzen
- Wissensbasierte Entscheidung, computergestützte Entscheidung und Expertensysteme
- Anwendung der Systemtheorie in Technologie, Managementsysteme, Transportsysteme, biomedizinische Systeme
- Systeme zur flexiblen Fertigung.

Korrespondenz:
Prof. J. Swiatek
Technical University of Wroclaw
Institute of Control and Systems Engineering
Janiszewski Str. 11—17
50-370 Wroclaw
Poland

LCA '89

Vom 8. bis 10. November wird in Mailand/Italien das 2. IFAC-Symposium „Low Cost Automation“ (LCA '89) durchgeführt. Die wissenschaftliche Organisation der Tagung liegt in den Händen des „Department of Computer and Systems Science“ der Universität Rom. Die Veranstaltung steht unter der Schirmherrschaft der IFAC und der „Associazione Nazionale Italiana per l'Automazione“ (ANIPLA).
Das 1. Symposium „Low Cost Automation“ fand 1986 in Valencia/Spanien statt und stieß auf reges Interesse der akademischen und der industriellen Welt.
„Low Cost Automation“ beinhaltet geeignete Entwurfstechniken für Steuerstrategien und Geräte für die Implementierung. Diese Geräte sind technisch weit fortgeschritten und einfach in Anlagen und Produktionsprozessen einsetzbar.

Schwerpunkte der LCA '89:

- Steuerungs- und Regelungstechniken
- Komponenten und Instrumentierung
- Anwendungen
- Ausbildung
- Soziale Auswirkungen.

Korrespondenz:
IFAC-LCA '89 Secretariat
Dip. Informatica e Sistematistica
Via Eudossiana, 18
I 00184 Roma
Italia

Neue Fachsektion der Gesellschaft für Informatik der DDR

Am 1. und 2. Dezember 1988 fand in Suhl die erste Jahrestagung der neu konstituierten Fachsektion „Informatik und Gesellschaft“ der Gesellschaft für Informatik der DDR zum Thema „Computer und Gesellschaft“ statt.
Die Fachsektion stellt sich die Aufgabe, einem breiten Kreis von Interessenten neue wissenschaftliche Erkenntnisse zur Wechselbeziehung von Informatik und Gesellschaft zugänglich zu machen. Sie fördert das Zusammenwirken von Instituten und Wissenschaftlern, unterstützt den Austausch von Forschungsergebnissen und den wissenschaftlichen Meinungsstreit zu gesellschaftlichen Voraussetzungen und Konsequenzen neuer Informations- und Kommunikationstechnologien sowie zu Konzepten und Methoden ihrer bewußten Gestaltung und erarbeitet dazu eigene Standpunkte. Das Leitungsgremium besteht aus Prof. Dr. sc. oec. Wolfgang Uhr, TU Dresden, Sektion Sozialistische Betriebswirtschaft, Vorsitzender der Fachsektion, Prof. Dr. phil. habil. Klaus Fuchs-Kittowski, Humboldt-Universität zu Berlin, Sektion Wissenschaftstheorie und -organisation, stellvertretender Vorsitzender, Prof. Dr. oec. habil. Gerd Friedrich, Stellvertretender Direktor des Zentralinstituts für Sozialistische Wirtschaftsführung beim ZK der SED, Berlin, sowie den Leitern der Arbeitsgruppen und Dr. oec. Dipl.-Math. Horst Bürdek, TU Dresden, Sekretär der Fachsektion. Die Fachsektion besteht aus folgenden sechs Arbeitsgruppen (AG):
AG 1: Rechnerunterstützte Betriebswirtschaft/Informatik und Wirtschaftswissenschaften (Leitung: Prof. Dr. sc. oec. W. Uhr, TU Dresden, Sektion Sozialistische Betriebswirtschaft)
AG 2: Informatik und Recht (Leitung: Prof. Dr. sc. jur. R. Osterland, TU Dresden)
AG 3: Komplexe nutzerorientierte Gestaltung von Informationssystemen und Mensch-Maschine-Beziehung (Leitung: Prof. Dr. phil. habil. K. Fuchs-Kittowski, Humboldt-Universität zu Berlin, Sektion Wissenschaftstheorie und -organisation)
AG 4: Informatik und Weltanschauung (Leitung: Prof. Dr. habil. L. Striebing, TU Dresden)
AG 5: Informatik und Kultur (Leitung: Prof. Dr. habil. H. Völz, AdW der DDR)
AG 6: Informatik und globale Probleme (Leitung: Prof. Dr. sc. phil. Dr.-Ing. M. Roth, TH Ilmenau).

ISSN 0026-0347, mess., steuern, regeln
Berlin 32 (1989) 5, S. 193—240

Neuerscheinungen und Nachauflagen

Angewandte Akustik Band 3

Herausgegeben von Prof. Dr.-Ing. habil. Wolfgang Kraak und Prof. Dr. sc. techn. Günther Schommartz. Reihe Angewandte Akustik. 1. Auflage. 216 Seiten, 125 Bilder, 25 Tafeln, Broschur, DDR 27,— M, Ausland 32,— DM. Bestellangaben: 554 059 6/ Angewandte Akustik Bd. 3

Die Buchreihe „Angewandte Akustik" berichtet in Übersichten und Beiträgen monographischen Charakters über aktuelle Probleme, Fortschritte und Tendenzen auf allen Gebieten der Akustik. Sie dient damit auch einer Integration der in vielen Disziplinen arbeitenden Akustiker.

Band 3 enthält u. a. folgende Beiträge:

Ultraschalldiagnostik — physikalische Grundlagen und prinzipielle Anwendungen · Digitale Signalanalyse in der Hydroakustik (Teil 2) · Signalverarbeitung in Spracheingabe- und Signalausgabesystemen · Schallemissionsanalyse — eine Methode der technischen Diagnostik · Meßtechnische Erfassung der auditiven Räumlichkeit · Digitaltechnik in der Tonstudio- und Beschallungstechnik · Signalverarbeitungsprinzipien auf der Basis von akustischen Oberflächenwellen.

Konstruktionselemente der Feinmechanik

Herausgegeben von Prof. Dr.-Ing. habil. Werner Krause. 1. Auflage. 752 Seiten, 985 Bilder, 320 Tafeln, Leinen, DDR 56,— M, Ausland 110,— DM. Bestellangaben: 553 866 4/Krause, Feinmechanik

Dieses neue Lehr- und Nachschlagewerk berücksichtigt die mit dem Einsatz der Mikroelektronik in der Gerätetechnik entstandenen erhöhten Ansprüche an die Feinmechanik. Nach einer Charakterisierung dieser Elemente werden zunächst rationelle Methoden zu deren konstruktiven Entwicklung dargestellt. Dazu gehören allgemeingültige Entwurfs- und Gestaltungsrichtlinien sowie der Rechnereinsatz, in TGL-Standards und DIN-Normen festgelegte Arbeitsunterlagen, die Statik, Dynamik, Festigkeitslehre und die Konstruktionswerkstoffe. Die Hauptkapitel umfassen darauf aufbauend dann mechanische Verbindungselemente und elektrische Leitungsverbindungen sowie mechanische Funktionselemente. Ein abschließender Abschnitt zur Mikromechanik zeigt darüber hinaus zukunftsträchtige Möglichkeiten zur Herstellung extrem miniaturisierter mechanischer Funktionselemente mit modernen Verfahren der Elektroniktechnologie.

Berechnungsbeispiele am Ende der jeweiligen Hauptabschnitte ermöglichen ein vertieftes Einarbeiten in komplizierte Sachverhalte.

VEM-Handbuch Relaisschutztechnik

Von Obering. Wolfgang Doemeland. Herausgeber: Zentrum für Forschung und Technologie des VEB Elektroprojekt und Anlagenbau Berlin. 4., bearbeitete Auflage. Etwa 368 Seiten, 284 Bilder, 78 Tafeln, Kunstleder, DDR etwa 27,50 M, Ausland etwa 38,— DM. Bestellangaben: 553 893 9/Doemeland, Relaisschutz

Behandelt werden Einrichtungen zur Meßgrößenerfassung (Wandler) und -verarbeitung (Schutzrelais) sowie zur Kommandoweiterleitung (Steuer- und Meldestromkreise). Zusammen mit dem Betriebsverhalten von Motoren, Transformatoren, Leitungen und Generatoren werden die Einstellung, Prüfung, Inbetriebsetzung und Wartung der Relaisschutzeinrichtungen beschrieben. Hinweise für den Arbeitsschutz, die technische Sicherheit und Protokollführung werden gegeben.

Die 4. Auflage berücksichtigt u. a. neue Vorschriften, veränderte Bewertungskriterien beim Einsatz von Stromwandlern, den Schutz von Netzen mit niederohmiger und kurzzeitig niederohmiger Sternpunkterdung sowie Aspekte beim Einsatz von Schutzrelais mit elektronischer Schaltungstechnik.

Russische Abkürzungen

Zusammengestellt von Edgar Scheitz. 2., unveränderte Auflage. 696 Seiten, Leinen, DDR 50,— M, Ausland 70,— DM. Bestellangaben: 553 225 4/Scheitz, Abkürzungen

Nachschlagewerk mit etwa 40 000 russischsprachigen Abkürzungen mit den entsprechenden russischsprachigen Klartexten und Begleitinformationen aus allen Wissensgebieten. Breiter Raum ist den Abkürzungen von staatlichen Einrichtungen und Betrieben der UdSSR gewidmet.

Auslieferung in diesen Tagen durch den Fachbuchhandel



VEB VERLAG TECHNIK BERLIN